



basic education

Department:
Basic Education
REPUBLIC OF SOUTH AFRICA

**NATIONAL
SENIOR CERTIFICATE**

GRADE 12

INFORMATION TECHNOLOGY P1

NOVEMBER 2013

MARKS: 120

TIME: 3 hours

This question paper consists of 16 pages and 3 annexures.

INSTRUCTIONS AND INFORMATION

1. The duration of this examination is three hours. Because of the nature of this examination it is important to note that you will not be permitted to leave the examination room before the end of the examination session.
2. No distinction has been made between the two programming languages in this question paper with regard to the formulation of the questions on programming. Where required, specific instructions have been provided for Delphi and Java candidates respectively.
3. You require the files listed below in order to answer the questions. The invigilator/teacher will tell you where to find them.

Question1_Delphi:

Question1DB.mdb
Question1P.dpr
Question1P.res
Question1U.dfm
Question1U.pas
tblDanceCouples.txt
tblResults.txt

Question1_Java:

Question1.java
Question1DB.mdb
tblDanceCouples.txt
tblResults.txt
TestQuestion1.java

Question2_Delphi:

Question2P.dpr
Question2P.res
Question2U.dfm
Question2U.pas
uDanceCouple.pas

Question2_Java:

DanceCouple.java
TestQuestion2.java

Question3_Delphi:

DataQ3.txt

Question3_Java:

DataQ3.txt
MenuQ3_Java.txt

If you received the files above on an external medium (CD, stiffy or flash drive) write your examination number on the label.

4. Type your examination number as a comment in the first line of each program file that contains your programming code.
5. Your program should always be coded to answer the question in such a way that it will run with different sets of input data.
6. Read ALL the questions carefully. Do not do more than the questions require.
7. Read the entire question before you answer any subquestions.
8. Save your work at regular intervals as a precaution against power failures.

9. During the examination, you may use the manuals originally supplied with the hardware and software. You may also use the HELP functions of the software. Java candidates may use the Java API files. You may NOT use any other resource material.
10. At the end of this examination session, you must hand in the external medium with all your work saved on it OR you must make sure that all your work has been saved on the network as explained to you by the invigilator/teacher.
11. Ensure that all files saved on the external medium or network can be read.
12. If required, make printouts of the programming code for all of the questions you have done.
13. All printing of the questions that you have done will take place within an hour of the completion of this examination.

SCENARIO

Your community has planned different events to raise funds for the local community centre. One event planned is an annual dance competition.

QUESTION 1: PROGRAMMING AND DATABASE

The dance competition is held over a period of 12 weeks. It starts with 14 competing dance couples. Some of the competitors are professional dancers. Each week one of the dance couples is eliminated based on their performance at the weekly elimination round. During the final week of the competition two elimination rounds will take place to determine who the winning dance couple will be.

A Microsoft Office Access database named **Question1DB.mdb**, two text files (**tblDanceCouples.txt** and **tblResults.txt**) and an incomplete program are given in the folder named **Question1_XXXX**, where XXXX refers to the programming language you have studied.

The design of the tables in the **Question1DB** database and sample data from each table are given in **ANNEXURE A**.

Do the following:

- Make a backup copy of the **Question1DB** database BEFORE you start answering the questions. You will need a copy of the original database to be able to test your program thoroughly.
- Rename the folder for QUESTION 1 by replacing the name of the programming language you have studied with your examination number.
- Open the incomplete program for QUESTION 1.
- Enter your examination number as a comment in the first line of the program file.
- Compile and execute the program. The interface displays eight menu options: Option A to Option G and a 'Quit' option.

NOTE:

- An error message will be displayed if any of Option A to Option G are selected because of the incomplete SQL statements.
- If you experience any problems using the database or connecting to the database, refer to **ANNEXURE B (Delphi)/ANNEXURE C (Java)** for troubleshooting hints.
- If you still experience database problems, you must nevertheless do the SQL code and submit it for marking. **Marks will only be awarded for the programming code that contains the SQL statements.**

- Complete the code for each menu option by formulating an appropriate SQL statement to display the respective query results as described in QUESTIONS 1.1 to 1.7 below.

NOTE: The code to some input statements and the code to execute the SQL statements and display the results of the queries have already been written as part of the given code.

1.1 Menu Option A

Display all the information in the **tblResults** table. Sort the data alphabetically according to the **TypeOfDance** field and then according to the **RoutineNo** field in descending order.

Example of the output of the first four records:

RoutineNo	Week	Round	DanceCoupleID	TypeOfDance	Song	Score	Result
116	12	2	8	American Smooth	Can't Help Falling in Love	39	Final
107	11	1	11	American Smooth	Time After Time	36	Safe
103	10	1	14	American Smooth	Singin' in the Rain	37	Safe
101	10	1	10	American Smooth	Pretty Woman	34	Safe

:

(3)

1.2 Menu Option B

Determine the dance routines with scores between 25 and 35 (including both) that were performed during the fifth week and the ninth week of the competition. Display the **RoutineNo**, **Week**, **TypeOfDance** and the **Score** for these routines.

Example of the output of the first four records:

RoutineNo	Week	TypeOfDance	Score
55	5	Tango	32
57	5	Paso Doble	31
58	5	American Smooth	35
60	5	Tango	34

:

(4)

1.3 Menu Option C

Allow the user to enter the name of a type of dance, for example Rumba.

Count and display the number of times this type of dance was performed during the competition. Name the calculated field **NumberOfPerformances**. Display the **TypeOfDance** and the **NumberOfPerformances**.

Example of the output if Rumba were entered by the user:

TypeOfDance	NumberOfPerformances
Rumba	7

(5)

1.4 **Menu Option D**

Determine the titles of the songs starting with the word 'Love' or containing the word 'you' or variations of the word 'you' that professional dance couples performed their dance routines to. Display the titles of the songs and the names of the dance couples who performed routines to these songs.

Example of the output:

Song	DancePartner1	DancePartner2
Love Man	Robbie	Ola
Love Ain't Here Anymore	Robbie	Ola
Let Me Entertain You	Robbie	Ola
You Sexy Thing	Robbie	Ola
You Can't Stop the Beat	Anita	Robin
I've Got You Under My Skin	Anita	Robin
Love Potion No. 9	Chelsee	Pasha
Spice Up Your Life	Chelsee	Pasha
Because of You	Chelsee	Pasha

(7)

1.5 **Menu Option E**

Calculate and display the average score for each dance couple in a calculated field named **AverageScore**.

The average score for each dance couple must be calculated using the total score and the number of dance routines they performed. Display the average score, correct to three decimal places. Display the **DanceCoupleID** and the **AverageScore** of all the dance couples.

Example of the output of the first four records:

DanceCoupleID	AverageScore
1	32.727
2	23.000
3	24.833
4	22.714

:

(6)

1.6 **Menu Option F**

Display a list with the names of all the dance couples who were eliminated up to the final week (week 12) of the competition. The names of each dance couple must appear only once on the list.

Example of the output (on the next page):

DancePartner1	DancePartner2
Alex	James
Anita	Robin
Audley	Natalie
Dan	Katya
Edwina	Vincent
Holly	Artem
Lulu	Brendan
Nancy	Anton
Robbie	Ola
Rory	Erin
Russell	Flavia

(5)

1.7 Menu Option G

Dance Couple 8 are announced as the winners of the competition. Update the content of the **Result** field for Dance Couple 8 to contain the word 'WINNERS' for the second round of the final week of the competition.

Once the records have been updated successfully, a suitable message will be displayed. The code for this message is supplied.

(5)

NOTE: If you want to test the menu options at this stage, use your backup copy of the **Question1DB** database.

- Enter your examination number as a comment in the first line of the file containing the SQL statements.
- Save your program.
- Make a printout of the code, if required.

[35]

QUESTION 2: OBJECT-ORIENTED PROGRAMMING

A program is needed to provide information about how the judges score the dance couples.

The files required for this question can be found in the folder named **Question2_XXXX**, where XXXX refers to the programming language you have studied.

You have been provided with an incomplete program that consists of:

Delphi	Java
<ul style="list-style-type: none"> • A class unit named uDanceCouple which describes the attributes of a dance couple and contains some methods 	<ul style="list-style-type: none"> • An object class named DanceCouple which describes the attributes of a dance couple and contains some methods
<ul style="list-style-type: none"> • A main form unit named Question2U 	<ul style="list-style-type: none"> • A test class named TestQuestion2

The given **uDanceCouple/DanceCouple** class contains the declaration and coding of:

- Three attributes describing a dance couple
- One constructor
- Three get/accessor methods

The three attributes describing a dance couple are:

Description	Names of attributes	
	Delphi	Java
The name of the first dance partner	fDanceP1	dance_p1
The name of the second dance partner	fDanceP2	dance_p2
The character A, B, C or D indicating the professional dance status of the couple	fProfessional	professional

NOTE: The professional dance status can be one of the following:

- A – Both dance partners are professional dancers.
- B – The first dance partner is a professional dancer.
- C – The second dance partner is a professional dancer.
- D – Neither of the dance partners is a professional dancer.

Any other character besides A, B, C or D will also be accepted and therefore the characters should not be validated.

Do the following:

- Rename the folder for QUESTION 2 by replacing the name of the programming language you have studied with your examination number.

Delphi	Java
<ul style="list-style-type: none"> • Open the incomplete project file Question2P.dpr. 	<ul style="list-style-type: none"> • Open the TestQuestion2.java test class file.
<ul style="list-style-type: none"> • Open the uDanceCouple.pas class unit. 	<ul style="list-style-type: none"> • Open the incomplete object class DanceCouple.java.
<ul style="list-style-type: none"> • Add your examination number as a comment in the first line of both the uDanceCouple.pas class unit and the Question2U.pas main form unit. 	<ul style="list-style-type: none"> • Add your examination number as a comment in the first line of both the DanceCouple.java object class and the TestQuestion2.java test class.

- Compile and execute the program. The interface displays four menu options: Option A, Option B, Option C and a 'Quit' option.

NOTE: Run the menu options in sequence when you test the program, in other words, first Option A, then Option B and then Option C.

2.1 Do the following to complete the code in the given **uDanceCouple/DanceCouple** class:

2.1.1 Write code for a **constructor** using parameters to initialise the three attributes of the class. The attributes are the names of the two dance partners and the professional dance status of the dance couple (see previous page). (5)

2.1.2 Write code for a method named **getWeighting** that will determine and return a weighting value for a dance couple based on the professional dance status of the dance couple. A higher weighting value is to the benefit of the dance couple. The following applies:

- If neither dance partner is a professional dancer, the weighting is set to 3.
- If one of the dance partners is a professional dancer, the weighting is set to 2.
- If both dance partners are professional dancers, the weighting is set to 1.
- If the professional dance status is any character other than A, B, C or D, the weighting is set to 0. (8)

- 2.1.3 Write code for a method named **calcFinalScore** that will receive an array that contains the scores of the four judges for a dance performance as a parameter. This method must calculate and return the final score for the dance routine.

The final score for a dance routine is calculated as follows:

- The scores of the **first** judge and the **third** judge are each multiplied by the weighting value of the dance couple, which is determined by their professional dance status.
- The scores of all four judges are then added up once the weighting value has been taken into account.

(7)

- 2.1.4 Write code for a **toString** method that will construct and return a string with labels and information about a dance couple object in the following format:

Dance couple: <dance partner 1> & <dance partner 2> Professional dance status: <character indicating the professional dance status of the dance couple>
--

Example of the output for Sarah and John with C as their professional dance status:

Dance couple: Sarah & John Professional dance status: C
--

(6)

- 2.2 Do the following to complete the code in the main form unit (Delphi)/test class (Java):

- 2.2.1 Write code to complete **Menu Option A** to do the following:

- Allow the user to enter the names of the dance couple – first the name of dance partner 1 and then the name of dance partner 2.
- Allow the user to enter the professional dance status (A, B, C or D) of the dance couple.

The professional dance status can be one of the following:

- A – Both dance partners are professional dancers.
- B – The first dance partner is a professional dancer.
- C – The second dance partner is a professional dancer.
- D – Neither of the dance partners is a professional dancer.

NOTE: Any other character can be accepted.

- Declare and instantiate a dance couple object using the user input values.
- Display the information for a dance couple as shown in the example below.

Example of the output if the names of the dance partners are Sarah and John and their professional dance status is C:

```
Dance couple: Sarah & John
Professional dance status: C
```

(8)

2.2.2 Write code to complete **Menu Option B** to do the following:

- Declare an array of integers to store the individual scores of the four judges for a dance routine. The scores have a possible maximum value of 10.

Example of the four possible scores: 8, 6, 7, 9

- Allow the user to enter the four scores one by one. Assign a value of zero to a score if an invalid score, for example a negative score or a score greater than 10, is entered.
- Call the relevant methods to calculate the final score for the dance routine and display the output in the following format:

```
Dance couple: <dance partner1> & <dance partner2>
Scores from the judges: <score1> <score2> <score3> <score4>
Weighting value: <weighting value>
Final score: <calculated final score>
```

Example of the output for the dance couple Sarah and John if the scores from the judges are 8, 6, 7 and 9:

```
Dance couple: Sarah & John
Scores from the judges: 8 6 7 9
Weighting value: 2
Final score: 45
```

(10)

2.2.3 Write code to complete **Menu Option C** to do the following:

Write the names of the dance couple and their final score to a new text file named **Score.txt** in the following format:

<dance partner 1>,<dance partner 2>,<final score>

Example of data in **Score.txt**:

Sarah, John, 45

Display a message indicating that the information has been written to a text file.

(7)

- Make sure that your examination number is entered as a comment in the first line of the class unit (Delphi)/object class (Java) as well as the main form unit (Delphi)/test class (Java).
- Save all the files.
- Make a printout of the code, if required. Print the class unit (Delphi)/object class (Java) and the main form unit (Delphi)/test class (Java).

[51]

QUESTION 3: PROBLEM-SOLVING PROGRAMMING

During the fifth week of the dance competition the organisers decide to allow members of the public to vote for their favourite dance couple. Each dance couple has a unique competition number. People who are 18 years or older can vote by sending an SMS with the number of their favourite dance couple. Four of the original 14 couples were already eliminated during previous rounds. The public can only vote for the dance couples who are still in the competition. The dance couple with the lowest number of votes from the public may be eliminated during this round.

The folder for QUESTION 3 contains a text file named **DataQ3.txt** for Delphi and two text files (**DataQ3.txt** and **MenuQ3.txt**) for Java.

Each line of text in the text file (**DataQ3.txt**) contains information of an SMS that was received in the following format:

```
<name of the person who voted>;<the person's age in years>#<cellphone number>;<the competition number of the dance couple the person voted for>
```

Example of the data for the first three SMSs in the text file named **DataQ3.txt**:

```
Armando Harris;56#+27848283872;8  
Luca Huonti;49#+2585514394;1  
Venere Laconi;47#+27768984873;12
```

The data of the first two lines of text can be interpreted as follows:

- Armando Harris, aged 56, used cellphone number +2784 828 3872 to vote for Dance Couple 8.
- Luca Huonti, aged 49, used cellphone number +258 551 4394 to vote for Dance Couple 1.

Do the following:

- Rename the folder for QUESTION 3 by replacing the name of the programming language you have studied with your examination number.
- Create a new program/project/application.
- Enter your examination number as a comment in the first line of the program file(s) you have created that will contain your code.
- Save the program file(s) using the question number as part of the filename in the renamed folder for QUESTION 3.

- Develop an interface as follows:

Delphi	Java
<ul style="list-style-type: none"> • Use a <i>MainMenu</i> component and develop an interface with three menu options: Option A, Option B and a 'Quit' option. 	<ul style="list-style-type: none"> • Copy the given code for displaying a menu from the MenuQ3.txt text file. The interface needs to display three menu options: Option A, Option B and a Quit option.
<ul style="list-style-type: none"> • Add a <i>RichEdit</i> component called redQ3 on the form to display information. Change the <i>Align</i> property to <i>aClient</i>. 	
<ul style="list-style-type: none"> • Use the following code to complete the 'Quit' menu option: Application.Terminate; 	<ul style="list-style-type: none"> • Use the following code to complete the 'Quit'-menu option: System.exit(0);

- Complete the code for each menu option as follows:

NOTE: Run the menu options in sequence when you test the program, in other words, first Option A and then Option B.

3.1 Menu Option A

Some of the SMSs captured in the text file are invalid. An SMS is invalid when:

- The vote was cast for a dance couple who have already been eliminated
- The vote was cast by a person who is younger than 18 years

Allow the user to enter the competition numbers of the four dance couples already eliminated from the competition. The user can enter any four values in the range from 1 to 14. Validate the values entered to comply with the following conditions:

- The value must be a number and not a character.
- The value must be inside the range of 1 to 14.

NOTE:

- The program may only continue once all four input values have been validated.
- The validated four numbers need to be available to be used in Menu Option B.

If the cellphone number starts with the prefix '+27', the vote is regarded as a vote from within the borders of South Africa. Otherwise it is regarded as an international vote.

Display a numbered list with the contact details of all persons who cast valid votes. Indicate if the person cast an international vote by displaying the phrase 'International vote' as part of his/her contact details as shown in the example below.

Display the total number of valid and invalid votes at the bottom of the list.

Example 1: An example of the output after Option A has been executed and Dance Couples 1, 2, 3 and 4 have been eliminated:

No.	Name	Cellphone number	
1	Armando Harris	+27848283872	
2	Venere Laconi	+27768984873	
3	Gabriella Thao	+27839161414	
:			
302	Lucas Herder	+2581101571	International vote
303	Giacobbe Spina	+2582751425	International vote
304	Rachel Delarosa	+27833054860	
Invalid votes: 196			
Valid votes: 304			

Example 2: An example of the output after Option A has been executed and Dance Couples 1, 5, 6 and 8 have been eliminated:

No.	Name	Cellphone number	
1	Venere Laconi	+27768984873	
2	Gabriella Thao	+27839161414	
3	Tim Anger	+2645519196	International vote
:			
272	Estotz Lizarazu	+27833247494	
273	Lucas Herder	+2581101571	International vote
274	Rachel Delarosa	+27833054860	
Invalid votes: 226			
Valid votes: 274			

(22)

3.2 Menu Option B

The organisers need a summarised report that lists the total number of valid votes that were cast for each dance couple. More than one dance couple may have the same number of votes. The dance couple(s) with the lowest number of valid votes may be eliminated.

Indicate the following in the report:

- A list with headings and subheadings with the total number of votes that were cast for each dance couple. Indicate the four dance couples who have already been eliminated.
- The dance couple(s) who may be eliminated next based on the least number of votes.

Write code to generate and display the required report as indicated in the following examples.

Example 1: An example of the output after Option A has been executed and Dance Couples 1, 2, 3 and 4 have been eliminated:

```
Votes received during week 5:
Dance couple      Votes
1                 (Eliminated)
2                 (Eliminated)
3                 (Eliminated)
4                 (Eliminated)
5                 30
6                 30
7                 30
8                 43
9                 27
10                25
11                25
12                33
13                25
14                36

Dance couple(s) who may be eliminated next: 10 11 13
```

Example 2: An example of the output after Option A has been executed and Dance Couples 1, 5, 6 and 8 have been eliminated:

```
Votes received during week 5:
Dance couple      Votes
1                 (Eliminated)
2                 31
3                 22
4                 20
5                 (Eliminated)
6                 (Eliminated)
7                 30
8                 (Eliminated)
9                 27
10                25
11                25
12                33
13                25
14                36

Dance couple(s) who may be eliminated next: 4
```

(12)

- Enter your examination number as a comment in the first line of the program.
- Save all the files.
- Make a printout of the code, if required.

[34]

TOTAL: 120

ANNEXURE A: DATABASE STRUCTURE AND SAMPLE DATA

This annexure shows the database structure and sample data for the tables contained in the **Question1DB.mdb** database used in **QUESTION 1**.

tblResults: This table contains data on the results of all the dance routines that were performed during the twelve weeks of the competition.

Field Name	Type	Size	Description
RoutineNo	Autonumber	Long Integer	A unique number for each dance routine that was performed during the competition
Week	Number	Integer	A number indicating the week during which the dance routine was performed. The competition runs over 12 weeks.
Round	Number	Integer	A number indicating the round during which the routine was performed. The final week has two rounds. All the other weeks have only one round.
DanceCoupleID	Number	Long Integer	A unique number to identify the dance couple who performed the dance item
TypeOfDance	Text	30	Type of dance that was performed, for example waltz, samba, cha-cha, et cetera
Song	Text	40	The title of the song used during the dance routine
Score	Number	Integer	The total score from the judges for the dance routine
Result	Text	15	Indicates the result after the dance routine was performed. The result can be one of the following: <ul style="list-style-type: none"> • Safe – The dance couple are still in the competition. • Eliminated – The dance couple have been eliminated. • Final – The dance couple are in the final week.

Example data:

RoutineNo	Week	Round	DanceCoupleID	TypeOfDance	Song	Score	Result
1	1	1	1	Cha-cha	Who's That Chick?	28	Safe
2	1	1	2	Waltz	Are You Lonesome Tonight?	24	Safe
3	1	1	3	Cha-cha	I've Got The Music in Me	17	Safe
4	1	1	4	Waltz	Angel	20	Safe
5	1	1	5	Cha-cha	Bad Boys	19	Safe
6	1	1	6	Waltz	Three Times a Lady	28	Safe
7	1	1	7	Cha-cha	Venus	21	Safe
8	1	1	8	Cha-cha	Moves Like Jagger	28	Safe

tblDanceCouples: This table contains the data of the 14 dance couples who took part in the dance competition.

Field Name	Type	Size	Description
DanceCoupleID	AutoNumber	Long Integer	A unique number assigned to each dance couple
DancePartner1	Text	10	The first name of dance partner 1
DancePartner2	Text	10	The first name of dance partner 2
ProfessionalDancers	Text	1	The symbol that indicates whether the dance partners are professional dancers. The following applies: <ul style="list-style-type: none"> • B – Both dance partners are professional dancers. • O – One of the dance partners is a professional dancer. • N – Neither dance partner is a professional dancer.
PreviousDancePartners	Yes/No		An indication whether the dance couple were dance partners in previous dance competitions

Example data:

DanceCoupleID	DancePartner1	DancePartner2	ProfessionalDancers	PreviousDancePartners
1	Holly	Artem	B	<input checked="" type="checkbox"/>
2	Dan	Katya	O	<input checked="" type="checkbox"/>
3	Lulu	Brendan	N	<input checked="" type="checkbox"/>
4	Audley	Natalie	O	<input type="checkbox"/>
5	Robbie	Ola	B	<input checked="" type="checkbox"/>
6	Anita	Robin	B	<input checked="" type="checkbox"/>
7	Russell	Flavia	N	<input type="checkbox"/>
8	Harry	Aliona	N	<input type="checkbox"/>
9	Rory	Erin	N	<input checked="" type="checkbox"/>
10	Alex	James	O	<input checked="" type="checkbox"/>
11	Chelsee	Pasha	B	<input type="checkbox"/>
12	Edwina	Vincent	O	<input type="checkbox"/>
13	Nancy	Anton	B	<input type="checkbox"/>
14	Jason	Kristina	N	<input type="checkbox"/>

ANNEXURE B: DELPHI – TROUBLESHOOTING DATABASE PROBLEMS

B.1 If you cannot use the database provided:

- Create your own database with the name **Question1DB** that includes two tables, named **tblResults** and **tblDanceCouples** in the same folder as your program for QUESTION 1.
- Import the two text files (**tblResults.txt** and **tblDanceCouples.txt**) to use as data for the different tables.
- The first line in the text files contains the field names to be used.

B.2 If your program cannot establish connectivity with the database, make sure that the database **Question1DB** is in the same folder as your program for QUESTION 1. If this is not the case, copy the database file into the same folder as your program.

B.3 If your program establishes connectivity with the database but no data is displayed:

- Click on the ADOQuery component named **qryRec**.
- Click on the ellipsis button (three dots) to the right of the 'ConnectionString' property in the Object Inspector.
- Click on the 'Build' button, which takes you to the Data Link Properties dialogue window.
- Click on the 'Provider' tab to open the 'Provider' tab sheet and select 'Microsoft Jet 4.0 OLE DB Provider'. Click on the 'Next' button.
- The 'Connection' tab sheet will be displayed. The first option on the 'Connection' tab sheet provides an ellipsis button (three dots) that allows you to browse and look for the **Question1DB** file. You will find this file in the **Question1** folder. Once you have found it, select the **Question1DB** file and then click on the 'Open' button.
- Remove the username 'Admin'.
- Click on the 'Test Connection' button.
- Click 'OK' on each of the open dialogue windows.

ANNEXURE C: JAVA – TROUBLESHOOTING DATABASE PROBLEMS

C.1 If you cannot use the given database:

- Create your own database with the name **Question1DB** that includes two tables, named **tblResults** and **tblCouples** in the same folder as your program for QUESTION 1.
- Import the two text files (**tblResults.txt** and **tblCouples.txt**) to use as data for the different tables.
- The first line in the text files contains the field names to be used.

C.2 If your program cannot establish connectivity with the database make sure that the database **Question1DB** is in the same folder as your program for QUESTION 1. If this is not the case, copy the database file into the same folder as your program.

C.3 If you cannot establish connectivity with the database with the given files, use the following source code to ensure database connectivity:

```
try
{
    Class.forName ('sun.jdbc.odbc.JdbcOdbcDriver');
    String filename = 'Question1DB.mdb';
    String database = 'jdbc:odbc:Driver={Microsoft Access Driver (*.mdb)};DBQ=';
        database += filename.trim () + ';DriverID=22;READONLY=true}';
    Connection conn = DriverManager.getConnection (database, '', '');
}
catch (Exception e)
{
    System.out.println ('Unable to connect with the database');
}
```



basic education

Department:
Basic Education
REPUBLIC OF SOUTH AFRICA

**NATIONAL
SENIOR CERTIFICATE**

GRADE 12

INFORMATION TECHNOLOGY P1

NOVEMBER 2013

MEMORANDUM

MARKS: 120

This memorandum consists of 34 pages.

GENERAL INFORMATION:

- These marking guidelines are to be used as the basis for the marking session. They were prepared for use by markers, all of whom are required to attend a rigorous standardisation meeting to ensure that the guidelines are consistently interpreted and applied in the marking of candidates' scripts.
- It is acknowledged that there may be different views about some matters of emphasis or detail in the guidelines, and different interpretations of the application thereof.
- Note that learners who provide an alternate correct solution to that given in the marking guidelines will have been given full credit for the relevant question.
- **Annexures A, B and C** (pages 3-6) include the marking grid for each question for using either one of the two programming languages.
- **Annexures D, E, F and G** (pages 7-19) contain the solutions for Delphi for QUESTIONS 1 to 3 in programming code.
- **Annexures H, I, J and K** (pages 20-32) contain the solutions for Java for QUESTIONS 1 to 3 in programming code.
- Copies of Annexures A, B and C should be made for each learner and completed during the marking session

ANNEXURE A**QUESTION 1: MARKING GRID – PROGRAMMING AND DATABASE****GENERAL NOTES:**

- Only penalise for the incorrect use of quotes ONCE. Repeated incorrect use of quotes in follow up questions doesn't get penalised.
- The use of = for strings, the use of LIKE may be used as alternative.

CENTRE NUMBER:		EXAMINATION NUMBER:	
QUESTION	DESCRIPTION	MAX. MARKS	LEARNER'S MARKS
1.1	<p>Query: Correct list of fields (or *)✓; correct table✓; ORDER BY correct fields in correct order✓</p> <p>SQL: SELECT * FROM tblResults ORDER BY TypeOfDance, RoutineNo Desc</p>	3	
1.2	<p>Query: Correct fields & table✓; WHERE Correct Score✓ both weeks✓ correct operator used (OR/IN) for the weeks✓</p> <p>SQL: SELECT RoutineNo, Week, TypeOfDance, Score FROM tblResults WHERE (Score BETWEEN 25 AND 35) AND (Week=5 OR Week=9)</p> <p>Alternative: (Score >= 25) AND (Score <= 35) (Score > 24) AND (Score < 36) Score IN [25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35] Week IN [5, 9]</p> <p>Score >= 25 And Score <= 35 And (Week = 5 Or Week = 9)</p> <p>(Score >= 25 And (Week = 5 Or Week = 9)) And (Score <= 35 And (Week = 5 Or Week = 9))</p> <p>Score >= 25 And Score <= 35 And Week IN (5,9)</p> <p>Score IN (25,26,27,28,29,30,31,32,33,34,35) Or Week IN (5,9)</p> <p>Score BETWEEN 25 And 35 and Week = 5 OR Score BETWEEN 25 And 35 and Week = 9</p> <p>NOTE: Check the correctness of alternative use of intervals</p>	4	

QUESTION 1: MARKING GRID – PROGRAMMING AND DATABASE – continue

1.3	<p>Query: Correct field & table✓; Count✓; AS NumberOfPerformances✓ WHERE TypeOfDance equals user input✓ GROUP BY TypeOfDance✓</p> <p>SQL: (D) SELECT TypeOfDance, Count(*) AS NumberOfPerformances FROM tblResults WHERE TypeOfDance = "+ sX + " GROUP BY TypeOfDance</p> <p>SQL: (J) SELECT TypeOfDance, Count(*) AS NumberOfPerformances FROM tblResults WHERE TypeOfDance = "+ sX + " GROUP BY TypeOfDance</p> <p>Alternative: May use Count(<field name>)</p> <p>NOTE: The use of Distinct is not allowed</p>	5	
1.4	<p>Query: Correct fields✓; both tables✓; WHERE linking tables on DanceCoupleID✓; professional dancers with AND operator✓; LIKE Love% ✓; OR-operators with correct use of brackets✓; LIKE %you%✓</p> <p>SQL: (D) SELECT Song, DancePartner1, DancePartner2 FROM tblDanceCouples, tblResults WHERE tblResults.DanceCoupleID = tblDanceCouples.DanceCoupleID AND (ProfessionalDancers = "B") AND ((Song Like "Love%") OR (Song LIKE "%you%"))</p> <p>SQL: (J) SELECT Song, DancePartner1, DancePartner2 FROM tblDanceCouples, tblResults WHERE tblResults.DanceCoupleID = tblDanceCouples.DanceCoupleID AND (ProfessionalDancers = 'B') AND ((Song Like 'Love%') OR (Song LIKE '%you%'))</p> <p>Alternative: Make use of ALIASES for table names Make use of INNER JOIN statement</p> <p>(D) The use of LEFT(Song, 4) = "Love" (J) The use of LEFT(Song, 4) = 'Love'</p> <p>NOTE: The use of * instead of % subtract only ONE mark</p>	7	

QUESTION 1: MARKING GRID – PROGRAMMING AND DATABASE – continue

1.5	<p>Query: Correct field & correct table✓; Format to THREE decimals✓; sum(Score)✓; divide by Count(*)✓; AS AverageScore✓; GROUP BY DanceCoupleID✓</p> <p>SQL: (D) SELECT DanceCoupleID, Format((Sum(Score)/Count(*)), "0.000") AS AverageScore FROM tblResults Group BY DanceCoupleID</p> <p>SQL: (J) SELECT DanceCoupleID, Format((Sum(Score)/Count(*)), '0.000') AS AverageScore FROM tblResults Group BY DanceCoupleID</p> <p>Alternative: Use of different formatting strings, e.g. "#.000"</p> <p>The use of AVG(Score) – TWO marks</p> <p>Round(<calculation>, 3) instead of Format</p>	6	
1.6	<p>Query: Correct fields✓ from both tables✓; WHERE linking tables on DanceCoupleID✓; Result equals Eliminated✓; No duplicates – check for use of week 12 (included must have DISTINCT/Group by) (less than 12 – no distinct/group by required)✓</p> <p>SQL: (D) SELECT DISTINCT DancePartner1, DancePartner2 FROM tblResults, tblDanceCouples WHERE (tblResults.DanceCoupleID = tblCouples.DanceCoupleID) AND (Result LIKE "Eliminated") AND (Week < 12)</p> <p>SQL: (J) SELECT DISTINCT DancePartner1, DancePartner2 FROM tblResults, tblDanceCouples WHERE (tblResults.DanceCoupleID = tblCouples.DanceCoupleID) AND (Result LIKE 'Eliminated') AND (Week < 12)</p> <p>Alternative: Make use of ALIASES for table names Alternative: make use of INNER JOIN statement</p> <p>NOTE: If week 12 is included then a DISTINCT/GROUP BY must be used.</p>	5	
1.7	<p>Query: UPDATE table✓; SET Result to WINNERS✓; WHERE Second Round✓; AND ✓; CoupleID is 8✓</p> <p>SQL: (D) UPDATE tblResults SET Result="WINNERS" WHERE Round = 2 AND DanceCoupleID = 8</p> <p>SQL: (J) UPDATE tblResults SET Result='WINNERS' WHERE Round = 2 AND DanceCoupleID = 8</p> <p>NOTE: The use of Week is optional</p>	5	
TOTAL:		35	

ANNEXURE B**QUESTION 2: MARKING GRID – OBJECT-ORIENTED PROGRAMMING****GENERAL NOTES:**

- If the learner changed any given data type (e.g. character to string) penalise with ONE mark.
- Syntax error (e.g. ;) penalise only ONCE.
- In Java the use of single = in stead of == penalise only ONCE.
- **NO USE OF OBJECT CLASS:** Q2.1.1: 0 marks; Q2.1.2: maximum of 6 marks; Q2.1.3: Maximum of 5 marks; Q2.1.4: maximum of 3 marks

CENTRE NUMBER:		EXAMINATION NUMBER:	
QUESTION	DESCRIPTION	MAX. MARKS	LEARNER'S MARKS
2.1.1	Parameterised CONSTRUCTOR All three correct parameters ✓ with correct data types ✓ Set dance partner name 1 ✓ set dance partner name 2 ✓ Set professional status ✓	5	
2.1.2	getWeighting METHOD: Correct method return type (integer) ✓; Structure of if/case/switch ✓ Testing for A and assign weighting value of 1 ✓; Testing for B and C and assign weighting value of 2 ✓ ✓ {using OR/multiple If's}; Testing for D and assign weighting value of 3 ✓; Assign a weighting value of 0 for an invalid dance status characters/default value of zero ✓; Return the content of the assigned variable ✓ NOTE: Accept return type string ONLY if conversion is used in call statements.	8	
2.1.3	calculateFinalScore METHOD: integer array parameter ✓; loop through array OR refer to individual elements in array ✓; call getWeighting method OR used weighting as an attribute of the class ✓; multiply array element 1 st ✓ and 3 rd by weighting ✓; add the scores ✓; return total score ✓	7	
2.1.4	toString METHOD: Correct method header and return type (string) ✓; Construct a line with correct label ✓; names of both partners with &-character ✓; Construct a line with professional dance status (call method/attribute) ✓ on next line ✓; Return string ✓	6	

QUESTION 2: MARKING GRID – OBJECT-ORIENTED PROGRAMMING – continue

2.2.1	MENU OPTION A: Prompt user for input✓; Input two strings for partner values✓; Input and use the first character of professional status✓; Declare object local/global in unit/class✓ and instantiate DanceCouple object✓; with values entered as arguments✓; in the correct order✓; Display the information of the DanceCouple object✓	8	
2.2.2	MENU OPTION B: Declare an array of integer to keep four scores✓; Get FOUR inputs from user✓; Validate FOUR input values in range 1 to 10✓; Assign a zero if value outside range✓; <i>1st line of output:</i> Display label✓ and call get methods for the names of partners/use global variables for names✓; <i>2nd line of output:</i> Display label and four values ✓ <i>3rd line of output:</i> Display label and call getWeighting method to display the weighting value ✓ <i>4th line of output:</i> Display label and call calcFinalScore method✓ with correct parameter ✓	10	
2.2.3	MENU OPTION C: Text file: ✓ {Delphi: AssignFile & Rewrite Java: Create object to use as a File writer} Construct line of text containing the names of both dance partners✓ and their final score✓ in comma separated format ✓; Write line of text to text file✓; CloseFile/Close writer object✓; Confirmation message to the user ✓ NOTE (Delphi): Accept the use of SaveToFile method of the richedit.	7	
	TOTAL:	51	

ANNEXURE C**QUESTION 3: MARKING GRID - PROBLEM-SOLVING PROGRAMMING****GENERAL NOTES:**

- Syntax error (e.g. ;) penalise only ONCE.

CENTRE NUMBER:		EXAMINATION NUMBER:	
QUESTION	DESCRIPTION	MAX. MARKS	LEARNER'S MARKS
3.1	<p>MENU OPTION A: Using appropriate data structure (1D/2D/object array) for all fourteen couples✓; Using a Loop until all user input valid✓: User input to indicate four couples eliminated✓; Validate each value to be in the correct range✓ Use VAL/try...except/catch to validate input not characters✓;</p> <p>Text file: {DELPHI: AssignFile, Reset JAVA: Create object to read from file} ✓; Loop through the lines of text in the file✓ Read line from text file✓ Split data (name✓, age✓, contact number✓, couple number✓) If valid age✓ AND dance couple not eliminated ✓ <i>valid vote:</i> Increment valid counter✓ If international vote✓ based on not +27✓ add label ✓ <i>NOT valid vote:</i> Increment invalid counter ✓ Display output concatenated line of info within the loop✓</p> <p>Display total valid votes✓; Display total invalid votes✓;</p>	22	
3.2	<p>MENU OPTION B: Using appropriate data structure(s) for counting votes✓</p> <p>Loop✓ to count votes for each couple ✓ (<i>can be done in question 3.1 inside the IF-statement for a valid vote</i>)</p> <p>Display headings and labels before loop✓ (<i>also accept as part of question 3.1</i>)</p> <p>Loop✓ to display votes received by each couple✓ or the label✓ "(Eliminated)" when couple is eliminated</p> <p>Initialise lowest value✓; determine lowest total votes✓ ignoring couples already eliminated✓ inside a loop OR sorting an array</p> <p>Display all the dance couple(s) who may be eliminated next✓ with label✓</p>	12	
TOTAL:		34	

SUMMARY OF LEARNER'S MARKS:

	QUESTION 1	QUESTION 2	QUESTION 3	GRAND TOTAL
MAX. MARKS	35	51	34	120
LEARNER'S MARKS				

ANNEXURE D: SOLUTION – QUESTION 1: DELPHI

```

unit Question1U_MEMO;
{A solution for Question 1}
interface

uses
  Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,
  Dialogs, StdCtrls, DB, ADO, ADOX, Grids, DBGrids, ExtCtrls, Buttons, Menus;

type
  TfrmQ1 = class(TForm)
    qryRec: TADOQuery;
    dsrQry: TDataSource;
    grdQ1: TDBGrid;
    mnuMain: TMainMenu;
    mnuOptionA: TMenuItem;
    mnuOptionB: TMenuItem;
    mnuOptionC: TMenuItem;
    mnuOptionD: TMenuItem;
    mnuOptionE: TMenuItem;
    mnuOptionF: TMenuItem;
    mnuOptionG: TMenuItem;
    mnuQuit: TMenuItem;
    procedure mnuOptionAClick(Sender: TObject);
    procedure mnuOptionBClick(Sender: TObject);
    procedure mnuOptionCClick(Sender: TObject);
    procedure mnuOptionDClick(Sender: TObject);
    procedure mnuOptionEClick(Sender: TObject);
    procedure mnuOptionFClick(Sender: TObject);
    procedure mnuOptionGClick(Sender: TObject);
    procedure mnuQuitClick(Sender: TObject);
  private
    { Private declarations }
  public
    { Public declarations }
  end;

var
  frmQ1: TfrmQ1;

implementation

{$R *.dfm}
//=====
procedure TfrmQ1.mnuOptionAClick(Sender: TObject);
begin
  qryRec.Close;
  qryRec.SQL.Text := 'SELECT * FROM tblResults ' +
                    'ORDER BY TypeOfDance, RoutineNo Desc';
  qryRec.Open;
end;
//=====
procedure TfrmQ1.mnuOptionBClick(Sender: TObject);
begin
  qryRec.Close;
  qryRec.SQL.Text := 'SELECT RoutineNo, Week, TypeOfDance, Score ' +
                    'FROM tblResults WHERE (Score BETWEEN 25 AND 35) AND ' +
                    '(Week=5 OR Week=9)';
  qryRec.Open;
end;
//=====
Copyright reserved

```

```

procedure TfrmQ1.mnuOptionCClick(Sender: TObject);
var
  sX : String;
begin
  sX := INPUTBOX('Question 1', 'Enter the TYPE of the dance e.g. Rumba.', '');
  qryRec.Close;
  qryRec.SQL.Text := 'SELECT TypeOfDance, Count(*) AS NumberOfPerformances '+
    'FROM tblResults WHERE TypeOfDance = "' + sX +
    '" GROUP BY TypeOfDance';

  qryRec.Open;
end;
//=====
procedure TfrmQ1.mnuOptionDClick(Sender: TObject);
begin
  qryRec.Close;
  qryRec.SQL.Text := 'SELECT Song, DancePartner1, DancePartner2 ' +
    'FROM tblCouples, tblResults '+
    'WHERE tblResults.DanceCoupleID=tblCouples.DanceCoupleID AND '+
    '(ProfessionalDancers = "B") AND '+
    '((Song Like "Love%") OR (Song LIKE "%you%"))';

  qryRec.Open;
end;
//=====
procedure TfrmQ1.mnuOptionEClick(Sender: TObject);
begin
  qryRec.Close;
  qryRec.SQL.Text := 'SELECT DanceCoupleID, ' +
    'Format((Sum(Score)/Count(*)), "0.000") AS '+
    'AverageScore FROM tblResults Group BY DanceCoupleID';

  qryRec.Open;
end;
//=====
procedure TfrmQ1.mnuOptionFClick(Sender: TObject);
begin
  qryRec.Close;
  qryRec.SQL.Text := 'SELECT DancePartner1, DancePartner2 ' +
    'FROM tblResults, tblCouples '+
    'WHERE (tblResults.DanceCoupleID=tblCouples.DanceCoupleID) '+
    'AND (Result LIKE "Eliminated") AND (Week < 12) ' +
    'GROUP BY DancePartner1, DancePartner2';

  qryRec.Open;
end;
//=====
procedure TfrmQ1.mnuOptionGClick(Sender: TObject);
begin
  qryRec.Close;
  qryRec.SQL.Text := 'UPDATE tblResults SET Result = "WINNERS" '+
    'WHERE Round = 2 AND DanceCoupleID = 8';

  qryRec.ExecSQL;
  MessageDlg('Records Processed Successfully', mtInformation, [mbok], 0);
end;
//=====
procedure TfrmQ1.mnuQuitClick(Sender: TObject);
begin
  Application.Terminate;
end;

end.

```

ANNEXURE E: SOLUTION – QUESTION 2: DELPHI**2.1. GIVEN DANCE COUPLE CLASS UNIT:**

```
unit uDanceCouple_MEMO;
{A solution for question 2 - class unit.}
interface

type

    TScoresArray = array[1..4] of integer;

    TDanceCouple = class(TObject)
    private
        fDancePartner1,
        fDancePartner2 : String;
        fProfessional : char;
    public
        constructor Create(); overload;
        constructor Create(sPartner1, sPartner2 : String; cProf : Char);
overload;
        function GetDancePartner1 : String;
        function GetDancePartner2 : String;
        function GetProfessional: char;
        function GetWeighting : integer;
        function CalcFinalScore(arrJudgeScores : TScoresArray) : integer;
        function toString : String;
    end;

implementation
{ TDanceCouple }

constructor TDanceCouple.Create();
begin

end;

constructor TDanceCouple.Create(sPartner1, sPartner2: String;
    cProf: char);
begin
    fDancePartner1 := sPartner1;
    fDancePartner2 := sPartner2;
    fProfessional := cProf;
end;

function TDanceCouple.GetDancePartner1: String;
begin
    Result := fDancePartner1;
end;

function TDanceCouple.GetDancePartner2: String;
begin
    Result := fDancePartner2;
end;

function TDanceCouple.GetProfessional : char;
begin
    Result := fProfessional;
end;

function TDanceCouple.GetWeighting: integer;
```

```

begin
  case fProfessional of
    'A' : Result := 1;
    'B', 'C' : Result := 2;
    'D' : Result := 3;
  else
    Result := 0;
  end; //case
end;

function TDanceCouple.CalcFinalScore(arrJudgeScores: TScoresArray): integer;
begin
  Result := (GetWeighting * arrJudgeScores[1]) + arrJudgeScores[2] +
    (GetWeighting * arrJudgeScores[3]) + arrJudgeScores[4];
end;

function TDanceCouple.toString: String;
begin
  Result := 'Couple: ' + GetDancePartner1 + ' & ' + GetDancePartner2 + #13 +
    'Professional dance status: ' + GetProfessional;
end;
end.

```

2.2 FORM UNIT – QUESTION 2:

```

unit Question2U_MEMO;
{A possible solution for question 2}

interface

uses
  Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,
  Dialogs, StdCtrls, ComCtrls, Menus, uDanceCouple_MEMO;

type
  TfrmQ2 = class(TForm)
    mnuMain: TMainMenu;
    mnuOptionA: TMenuItem;
    mnuOptionB: TMenuItem;
    mnuOptionC: TMenuItem;
    mnuQuit: TMenuItem;
    redQ2: TRichEdit;
    procedure mnuQuitClick(Sender: TObject);
    procedure mnuOptionAClick(Sender: TObject);
    procedure mnuOptionBClick(Sender: TObject);
    procedure mnuOptionCClick(Sender: TObject);
  private
    { Private declarations }
  public
    { Public declarations }
  end;

var
  frmQ2: TfrmQ2;
  DanceCouple : TDanceCouple;
  arrScores : TScoresArray;

implementation
{$R *.dfm}
{$R+}

```



```

procedure TfrmQ2.mnuOptionAClick(Sender: TObject);
var
  sPart1, sPart2 : String;
  cProf : char;
begin
  {Code Option A}
  sPart1 := InputBox('Partner 1', 'Enter the name of partner 1', 'Sarah');
  sPart2 := InputBox('Partner 2', 'Enter the name of partner 2', 'John');
  cProf := InputBox('Dance status', 'Enter the professional dance status of
the couple [A, B, C, D]', 'C')[1];
  DanceCouple := TDanceCouple.Create(sPart1, sPart2, cProf);
  redQ2.Lines.Add(DanceCouple.toString);
end;

procedure TfrmQ2.mnuOptionBClick(Sender: TObject);
var
  k : integer;
begin
  {Code Option B}
  For k := 1 to 4 do
    begin
      arrScores[k] := StrToInt(InputBox('Scores', 'Enter the score of judge' +
IntToStr(k), ''));
      if (arrScores[k] > 10) OR (arrScores[k] < 0)
        then arrScores[k] := 0;
    end; //for
  redQ2.Lines.Clear;
  redQ2.Lines.Add('Couple: ' + DanceCouple.GetDancePartner1 + ' & ' +
DanceCouple.GetDancePartner2 );
  redQ2.Lines.Add('Scores from the judges: ' + IntToStr(arrScores [1]) + ' ' +
IntToStr(arrScores [2]) + ' ' +IntToStr(arrScores [3]) + ' ' +
IntToStr(arrScores [4]) );
  redQ2.Lines.Add('Weighting value: ' + IntToStr(DanceCouple.GetWeighting));
  redQ2.Lines.Add('Final score: ' +
IntToStr(DanceCouple.CalcFinalScore(arrScores)));
end;

procedure TfrmQ2.mnuOptionCClick(Sender: TObject);
var
  TFile : TextFile;
  sLine : String;
begin
  {Code Option C}
  redQ2.Lines.Clear;
  AssignFile(TFile, 'Score.txt');
  Rewrite(TFile);
  sLine := DanceCouple.GetDancePartner1 + ',' +
DanceCouple.GetDancePartner2 + ',' +
IntToStr(DanceCouple.CalcFinalScore(arrScores));
  Writeln(TFile, sLine);
  CloseFile(TFile);
  redQ2.Lines.Add('Data has been written to the file');
end;

procedure TfrmQ2.mnuQuitClick(Sender: TObject);
begin
  Application.Terminate;
end;

end.

```

ANNEXURE F: SOLUTION with OOP – QUESTION 3: DELPHI**3.1. COUPLE CLASS UNIT:**

```
unit uCouple;
//A solution for Question 3 - WITH OOP >> A Couple class
interface
type
  TCouple = class(TObject)
  private
    fNumber,
    fVotes      : integer;
    fEliminated : boolean;
  public
    constructor Create(iNumber : integer);
    function GetNumber : integer;
    procedure AddAVote;
    function GetVotes : integer;
    function GetEliminated : boolean;
    procedure SetEliminated(bEliminate : boolean);
    function toString : String;
  end;

implementation

uses SysUtils;

{ TCouple }

procedure TCouple.AddAVote;
begin
  Inc(fVotes , 1);
end;

constructor TCouple.Create(iNumber: integer);
begin
  fNumber := iNumber;
  fVotes  := 0;
  fEliminated := False;
end;

function TCouple.GetEliminated: boolean;
begin
  Result := fEliminated;
end;

procedure TCouple.SetEliminated(bEliminate: boolean);
begin
  fEliminated := bEliminate;
end;

function TCouple.GetNumber: integer;
begin
  Result := fNumber;
end;

function TCouple.GetVotes: integer;
begin
  Result := fVotes;
end;
```

```
function TCouple.toString: String;
begin
  IF GetEliminated
    then Result := IntToStr(fNumber) + #9 + '(Eliminated)'
    else Result := IntToStr(fNumber) + #9 + IntToStr(fVotes);
end;

end.
```

3.2 A VOTER CLASS UNIT:

```
unit uVoter;
// A solution for Question 3 - WITH OOP >> A Voter class
interface

type
  TVoter = class(TObject)
  private
    fName,
    fContact      : String;
    fAge          : integer;
  public
    constructor Create(sName, sContact : String; iAge : integer);
    function GetAge      : integer;
    function isInternational:boolean;
    function toString   : String;
  end;

implementation

{ TVoter }

constructor TVoter.Create(sName, sContact: String; iAge : integer);
begin
  fName      := sName;
  fContact  := sContact;
  fAge      := iAge;

end;

function TVoter.GetAge: integer;
begin
  Result := fAge;
end;

function TVoter.isInternational:boolean;
begin
  Result := (Copy(fContact, 1, 3) <> '+27');
end;

function TVoter.toString: String;
begin
  IF isInternational
    then Result := fName + #9 + fContact + #9 + 'International vote'
    else Result := fName + #9 + fContact;
end;

end.
```

3.3. FORM UNIT:

```

unit Question3U_OOP_MEMO;
// A solution for Question 3 with OOP
interface

uses
  Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,
  Dialogs, Menus, StdCtrls, ComCtrls,
  uVoter, uCouple; //user created classes

type
  TfrmQ3OOPMemo = class(TForm)
    redQ3: TRichEdit;
    mnuMain: TMainMenu;
    mnuOptionA: TMenuItem;
    mnuOptionB: TMenuItem;
    mnuQuit: TMenuItem;
    procedure mnuQuitClick(Sender: TObject);
    procedure mnuOptionAClick(Sender: TObject);
    procedure mnuOptionBClick(Sender: TObject);
  private
    { Private declarations }
  public
    { Public declarations }
  end;

var
  frmQ3OOPMemo: TfrmQ3OOPMemo;

implementation

{$R *.dfm}
{$R+}

var
  arrCouples : array[1..14] of TCouple;
  Eliminated : Set of 1..14;

procedure TfrmQ3OOPMemo.mnuQuitClick(Sender: TObject);
begin
  Application.Terminate;
end;

procedure TfrmQ3OOPMemo.mnuOptionAClick(Sender: TObject);
var
  AVoter : TVoter;
  TFile : TextFile;
  sLine, sName, sContact : String;
  iDataCount, iInvalidCount : integer;
  iAge, iCouple, iNumber, K : integer;
begin //menu Option A
  //create the 14 couples.
  for K := 1 to 14 do
    arrCouples[K] := TCouple.Create(K);

  //user input which couples were eliminated
  Eliminated := [];
  For K := 1 to 4 do
    begin
      try
        Repeat

```

```

        iNumber := StrToInt(InputBox('Eliminated Couple', 'Enter the number of
the couple (1..14)', '3'));
    Until (iNumber in [1..14]);
except
    on EConvertError do
    begin
        ShowMessage('Please enter a number.');
```

```

        Exit;
    end;
end; //try
Eliminated := Eliminated + [iNumber];
arrCouples[iNumber].SetEliminated(True);
end;

//Read data from txt file
AssignFile(TFile, 'DataQ3.txt');
Reset(TFile);
iDataCount := 0;
iInvalidCount := 0;
redQ3.Lines.Clear;
redQ3.Paragraph.TabCount := 3;
redQ3.Paragraph.Tab[0] := 40;
redQ3.Paragraph.Tab[1] := 200;
redQ3.Paragraph.Tab[2] := 300;
redQ3.Lines.Add('No.'+#9+'Name'+#9+'Contact number');
```

```

While NOT EOF(TFile) DO
begin
    Readln(TFile, sLine);
    //split data >> name age contact & couple number
    sName := Copy(sLine, 1, Pos(';', sLine)-1);
    Delete(sLine, 1, Pos(';', sLine));
    iAge := StrToInt(Copy(sLine, 1, Pos('#', sLine)-1));
    Delete(sLine, 1, Pos('#', sLine));
    sContact := Copy(sLine, 1, Pos(';', sLine)-1);
    Delete(sLine, 1, Pos(';', sLine));
    iCouple := StrToInt(sLine);

    AVoter := TVoter.Create(sName, sContact, iAge);
    IF (AVoter.GetAge >= 18) and NOT(iCouple IN Eliminated)
    then
    begin
        Inc(iDataCount, 1);
        redQ3.Lines.Add(IntToStr(iDataCount) + #9 + AVoter.toString);
        arrCouples[iCouple].AddAVote;
    end
    else
    begin
        Inc(iInvalidCount, 1);
    end;
end; //while
CloseFile(TFile);
redQ3.Lines.Add(' ');
redQ3.Lines.Add('Invalid votes: ' + IntToStr(iInvalidCount) + #13 +
'Valid votes: ' + IntToStr(iDataCount));
AVoter := nil;
end;

procedure TfrmQ3OOPMemo.mnuOptionBClick(Sender: TObject);
var
    sLine : String;
    K, iLow : integer;

```

```
begin
// Option B
redQ3.Lines.Clear;
redQ3.Paragraph.TabCount := 3;
redQ3.Paragraph.Tab[0] := 100;
redQ3.Paragraph.Tab[1] := 200;
redQ3.Paragraph.Tab[2] := 300;
sLine := '';
for K := 1 to 14 do
  IF arrCouples[K].GetEliminated
  then sLine := sLine + IntToStr(arrCouples[K].GetNumber) + ' ';

redQ3.Lines.Add('Votes received during week 5:');
redQ3.Lines.Add('Couple' + #9 + 'Votes');
for K := 1 to 14 do
  redQ3.Lines.Add(arrCouples[K].toString);

//determine the lowest total votes
iLow := 500;
For K := 1 to 14 do
  IF (arrCouples[K].GetVotes < iLow) AND (arrCouples[K].GetVotes <> 0)
  then
  begin
    iLow := arrCouples[K].GetVotes;
  end;

redQ3.Lines.Add(' ');
sLine := '';
For K:= 1 to 14 do
  IF (arrCouples[K].GetVotes = iLow)
  then
    sLine := sLine + IntToStr(arrCouples[K].GetNumber) + ' ';

  redQ3.Lines.Add('Couple(s) who may be eliminated next: ' + sLine);
end;

end.
```

ANNEXURE G: SOLUTION without OOP– QUESTION 3: DELPHI

```

unit Question3U_MEMO;
//A solution for question 3 without using OOP.
interface

uses
  Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,
  Dialogs, Menus, StdCtrls, ComCtrls;

type
  TfrmQuest3 = class(TForm)
    mnuMain: TMainMenu;
    mnuOptionA: TMenuItem;
    mnuOptionB: TMenuItem;
    mnuQuit: TMenuItem;
    redQ3: TRichEdit;
    procedure mnuQuitClick(Sender: TObject);
    procedure mnuOptionAClick(Sender: TObject);
    procedure mnuOptionBClick(Sender: TObject);
  private
    { Private declarations }
  public
    { Public declarations }
  end;

var
  frmQuest3: TfrmQuest3;

implementation

{$R *.dfm}
{$R+}

VAR
  iDataCount,
  iInvalidCount : integer;
  arrCouples    : array[1..14] of integer;
  Eliminated    : Set of 1..14;

procedure TfrmQuest3.mnuOptionAClick(Sender: TObject);
var
  TFile           : TextFile;
  sLine, sName, sContact : String;
  iAge, iCouple, iNumber, K : integer;
begin
  //user input - eliminated couples
  Eliminated := [];
  For K := 1 to 14 do
    arrCouples[K] := 0;
  For K := 1 to 4 do
    begin
      try
        Repeat
          iNumber := StrToInt(InputBox('Eliminated couple', 'Enter the number of
the couple (1 - 14)', '3'));
        Until (iNumber in [1..14]);
      except
        on EConvertError do
          begin
            ShowMessage('Please type in a number.');
```

```

        end;
    end; //try
    Eliminated := Eliminated + [iNumber];
end;

//Read data from text file
AssignFile(TFile, 'DataQ3.txt');
Reset(TFile);
iDataCount := 0;
iInvalidCount := 0;
redQ3.Lines.Clear;
redQ3.Paragraph.TabCount := 3;
redQ3.Paragraph.Tab[0] := 40;
redQ3.Paragraph.Tab[1] := 200;
redQ3.Paragraph.Tab[2] := 300;
redQ3.Lines.Add('No.'+#9+'Name '+#9+'Cellphone number');

While NOT EOF(TFile) DO
begin
    Readln(TFile, sLine); {x}
    //split data >> name age contact & couple number
    sName := Copy(sLine, 1, Pos(';', sLine)-1);
    Delete(sLine, 1, Pos(';', sLine));
    iAge := StrToInt(Copy(sLine, 1, Pos('#', sLine)-1));
    Delete(sLine, 1, Pos('#', sLine));
    sContact := Copy(sLine, 1, Pos(';', sLine)-1);
    Delete(sLine, 1, Pos(';', sLine));
    iCouple := StrToInt(sLine);

    //validate >> Age>=18 couple not eliminated
    if (iAge >= 18) AND NOT(iCouple IN Eliminated)
    then
        begin //valid vote
            Inc(iDataCount, 1);
            Inc(arrCouples[iCouple], 1);

            sLine := IntToStr(iDataCount) + #9 + sName + #9 + sContact;
            if Copy(sContact, 1, 3) <> '+27' //International vote
            then
                sLine := sLine + #9 + 'International vote';

            redQ3.Lines.Add(sLine);
        end
    else
        begin
            //Invalid vote >> discard vote.
            Inc(iInvalidCount, 1);
        end; //else
    end; //while
CloseFile(TFile);
redQ3.Lines.Add(' ');
redQ3.Lines.Add('Invalid votes: ' + IntToStr(iInvalidCount) + #13 +
                'Valid votes: ' + IntToStr(iDataCount));
end;

procedure TfrmQuest3.mnuOptionBClick(Sender: TObject);
var
    K, iCLow : integer;
    sLine : String;
begin
    //Option B
    redQ3.Lines.Clear;

```



```
redQ3.Paragraph.TabCount := 3;
redQ3.Paragraph.Tab[0] := 100;
redQ3.Paragraph.Tab[1] := 200;
redQ3.Paragraph.Tab[2] := 300;
sLine := '';

redQ3.Lines.Add('Votes received during week 5:');
redQ3.Lines.Add('Couple' +#9 + 'Votes');
for K := 1 to 14 do
begin
  if K in Eliminated
  then redQ3.Lines.Add(IntToStr(K) + #9 + '(Eliminated)')
  else redQ3.Lines.Add(IntToStr(K) + #9 + IntToStr(arrCouples[K]));
end;

//Calculate lowest number of votes
iCLow := 500;
For K := 1 to 14 do
  IF (arrCouples[K] < iCLow) AND (arrCouples[K] > 0)
  then iCLow := arrCouples[K];

//Display couples who may be eliminated next
redQ3.Lines.Add(' ');
sLine := '';
For K := 1 to 14 do
  IF (arrCouples[K] = iCLow)
  then
    sLine := sLine + IntToStr(K) + ' ';

redQ3.Lines.Add('Couple(s) who may be eliminated next: ' + sLine);
end;

procedure TfrmQuest3.mnuQuitClick(Sender: TObject);
begin
  Application.Terminate;
end;

end.
```

ANNEXURE H: SOLUTION – QUESTION 1: JAVA

```
// A solution for Question 1

import java.io.BufferedReader;
import java.io.InputStreamReader;
import java.io.IOException;
import java.sql.*;
import java.util.Scanner;

public class TestQuestion1
{
    public static void main(String[] args) throws SQLException, IOException
    {
        Scanner sc = new Scanner(System.in);
        // OR
        // BufferedReader inKb = new BufferedReader(new
InputStreamReader(System.in));

        Question1 DB = new Question1();
        System.out.println();

        char choice = ' ';
        do {
            System.out.println("\n\n      MENU");
            System.out.println();
            System.out.println("      Option A");
            System.out.println("      Option B");
            System.out.println("      Option C");
            System.out.println("      Option D");
            System.out.println("      Option E");
            System.out.println("      Option F");
            System.out.println("      Option G");
            System.out.println();
            System.out.println("      Q - QUIT");
            System.out.println(" ");
            System.out.print("      Your choice? ");
            choice = sc.nextLine().toUpperCase().charAt(0);
            //OR
            //choice = inKb.readLine().toUpperCase().charAt(0);
            System.out.println(" ");
            String sql = "";
            switch (choice) {
                case 'A': // Question 1.1
                    {
                        sql = "SELECT * FROM tblResults ORDER BY TypeOfDance,
RoutineNo Desc";
                        DB.query(sql);
                        break;
                    }

//=====
                case 'B': // Question 1.2
                    {
                        sql = "SELECT RoutineNo, Week, TypeOfDance, Score FROM
tblResults WHERE (Score BETWEEN 25 AND 35) AND (Week=5 OR Week=9)";
                        DB.query(sql);
                        break;
                    }

//=====

```

```

        case 'C': // Question 1.3
        {
            System.out.println("Question 1: Enter the TYPE of the
dance e.g. Rumba.");
            String sX = sc.nextLine();
                                //OR
                                // String sX = inKb.readLine();
            sql = "SELECT TypeOfDance, Count(*) AS
NumberOfPerformances FROM tblResults WHERE TypeOfDance = '"+ sX + "' GROUP BY
TypeOfDance";
            DB.query(sql);
            break;
        }

//=====
        case 'D': // Question 1.4
        {
            sql = "SELECT Song, DancePartner1, DancePartner2 FROM
tblCouples, tblResults WHERE tblResults.DanceCoupleID=tblCouples.DanceCoupleID
AND (ProfessionalDancers = 'B') AND ((Song Like 'love%') OR (Song LIKE
'%you%'))";
            DB.query(sql);
            break;
        }

//=====
        case 'E': // Question 1.5
        {
            sql = "SELECT DanceCoupleID,
Format((Sum(Score)/Count(*)), '0.000') AS AverageScore FROM tblResults Group BY
DanceCoupleID";
            DB.query(sql);
            break;
        }

//=====
        case 'F': // Question 1.6
        {
            sql = "SELECT DancePartner1, DancePartner2 FROM
tblResults, tblCouples WHERE tblResults.DanceCoupleID=tblCouples.DanceCoupleID
AND (Result LIKE 'Eliminated') AND (Week < 12) GROUP BY DancePartner1,
DancePartner2";
            DB.query(sql);
            break;
        }

//=====
        case 'G': // Question 1.7
        {
            sql = "UPDATE tblResults SET Result = 'WINNERS' WHERE
Round = 2 AND DanceCoupleID = 8";
            DB.query(sql);
            break;
        }
    } while (choice != 'Q');
    DB.disconnect();
    System.out.println("Done");
}
}
}

```

ANNEXURE I: SOLUTION – QUESTION 2: JAVA**2.1 COUPLE OBJECT CLASS:**

```
// POSSIBLE SOLUTION FOR QUESTION 2

public class DanceCouple
{
    private String dance_p1;
    private String dance_p2;
    private char professional;

    public DanceCouple()
    {
    }

    public DanceCouple(String part1, String part2, char pro)
    {
        dance_p1 = part1;
        dance_p2 = part2;
        professional = pro;
    }

    public String getDancePartner1()
    {
        return dance_p1;
    }

    public String getDancePartner2()
    {
        return dance_p2;
    }

    public char getProfessional()
    {
        return professional;
    }

    public int getWeighting()
    {
        int weighting = 0;
        if(professional == 'D')
            weighting =3;

        if(professional == 'B' || professional == 'C')
        {
            weighting = 2;
        }
        if(professional == 'A')
        {
            weighting = 1;
        }
        return weighting;
    }

    public int calcFinalScore(int[] scores)
    {
        int result = 0;
        for(int i = 0; i < scores.length; i++)
        {
```

```

        if(i == 0 || i == 2)
        {
            result += scores[i] * getWeighting();
        }
        else
        {
            result += scores[i];
        }
    }
    return result;
}

public String toString()
{
    return "Couple: " + getDancePartner1()+ " & " + getDancePartner2()
+ "\n" + "Professional dance status: " + getProfessional();
}
}

```

2.3 TEST/DRIVER CLASS:

```

// POSSIBLE SOLUTION FOR QUESTION 2
import java.io.*;
import javax.swing.*;
import java.text.*;
import java.util.Scanner;

public class TestQuestion2
{
    public static void main(String[] args) throws Exception
    {
        int[] scores = new int[4];
        DanceCouple couple = new DanceCouple();

        Scanner sc = new Scanner(System.in);
        // OR
        // BufferedReader inKb = new BufferedReader(new
InputStreamReader(System.in));

        char choice = ' ';
        do {
            System.out.println("  MENU\n");
            System.out.println("Option A");
            System.out.println("Option B");
            System.out.println("Option C");
            System.out.println("");
            System.out.println("Q - QUIT");
            System.out.println("\nYour choice? ");
            choice = sc.nextLine().toUpperCase().charAt(0);
            //OR
            //choice = inKb.readLine().toUpperCase().charAt(0);
            switch (choice) {
                case 'A':
                    // OPTION A
                    System.out.println("Enter the name of partner 1 ");
                    String partner1 = sc.nextLine();
                    // OR
                    // String partner1 = inKb.readLine();
                    System.out.println("Enter the name of partner 2");
                    String partner2 = sc.nextLine();
                    // OR

```

```

        // String partner2 = inKb.readLine();
        System.out.println("Enter professional dance status of the
couple (A/B/C/D)");
        char professional = sc.nextLine().toUpperCase().charAt(0);
        // OR
        // char professional = inKb.readLine().toUpperCase().charAt(0);
        couple = new DanceCouple(partner1, partner2, professional);
        System.out.println(couple);
        System.out.println("");
        break;
    case 'B':
        // OPTION B
        System.out.println();
        for (int k = 0; k < 4; k++) {
            System.out.println("Enter the score from judge " + (k+1));
            scores[k] = Integer.parseInt(sc.nextLine());
            // OR
            // scores[k] = Integer.parseInt(inKb.readLine());
            if ((scores[k]>10)||((scores[k]<0))
            {
                scores[k]=0;
            }
        }
        System.out.println("Couple: " + couple.getDancePartner1() + "
& " + couple.getDancePartner2());
        System.out.println("Scores from the judges: " + scores[0] + "
" + scores[1] + " " + scores[2] + " " + scores[3]);
        System.out.println("Weighting value: " +
couple.getWeighting());
        System.out.println("Final score: " +
couple.calcFinalScore(scores));
        System.out.println("");
        break;
    case 'C':
        // OPTION C
        BufferedWriter bw = new BufferedWriter(new
FileWriter("Score.txt"));
        bw.write(couple.getDancePartner1() + "," +
couple.getDancePartner2() + "," + couple.calcFinalScore(scores));
        bw.close();
        System.out.println("Data has been written to the text file");
        System.out.println("");
        break;
    case 'Q':
        System.out.println("QUIT");
    }
} while (choice != 'Q');
}
}

```

ANNEXURE J: SOLUTION with OOP – QUESTION 3: JAVA**3.1. COUPLE OBJECT CLASS:**

```
//POSSIBLE SOLUTION FOR QUESTION 3 WITH OOP
```

```
public class Couple {  
  
    private int num;  
    private int numVotes;  
    private boolean inComp; // couple still in competition  
  
    public Couple(int num, int numVotes) {  
        this.num = num;  
        this.numVotes = numVotes;  
        inComp = true;  
    }  
  
    public int getNum() {  
        return num;  
    }  
  
    public void setNum(int num) {  
        this.num = num;  
    }  
  
    public int getNumVotes() {  
        return numVotes;  
    }  
  
    public void setNumVotes(int numVotes) {  
        this.numVotes = numVotes;  
    }  
  
    public void addVote()  
    {  
        numVotes++;  
    }  
  
    public void setInComp()  
    {  
        inComp = false;  
    }  
  
    public boolean isInComp()  
    {  
        return inComp;  
    }  
  
    public String toString()  
    {  
        String output = "";  
        if(inComp)  
        {  
            output = String.format("%-15s%-10s", getNum(),getNumVotes());  
        }  
        else  
        {  
            output = String.format("%-15s%-10s", getNum() ,"(Eliminated)");  
        }  
    }  
}
```

```
        return output;
    }
}
```

3.2 A VOTER OBJECT CLASS:

```
public class Voter
{
    private String name;
    private String cellphone;
    private int age;
    private int vote;

    public Voter(String Name, String Cell, int Age, int Vote)
    {
        name = Name;
        cellphone =Cell;
        age = Age;
        vote = Vote;
    }

    public String getName() {
        return name;
    }

    public String getCellphone() {
        return cellphone;
    }
    public int getAge() {
        return age;
    }

    public int getVote() {
        return vote;
    }

    public boolean isInternational()
    {
        if(cellphone.substring(0,3).equals("+27"))
            return true;
        else
            return false;
    }

    public String toString()
    {
        String afvString = "";
        if(isInternational()){
            afvString = String.format("%-25s%-25s",getName(),getCellphone());
        }
        else
        {
            afvString = String.format("%-25s%-25s%-
25s",getName(),getCellphone(), "International vote");
        }
        return afvString;
    }
}
```


3.3 VotingProcess OBJECT CLASS:

```

import java.io.*;
import java.util.Scanner;

public class VotingProcess
{
    Couple[] couple = new Couple[14];
    public void readCouples() {
        for (int cnt = 0; cnt < 14; cnt++) {
            couple[cnt] = new Couple((cnt+1), 0);
        }
    }

    public void readFromFile() throws IOException
    {
        Scanner sf = new Scanner(new FileReader("DataQ3.txt"));
        // OR
        // BufferedReader vFile = new BufferedReader(new
FileReader("DataQ3.txt"));

        Scanner sc = new Scanner(System.in);
        // OR
        // BufferedReader kb = new BufferedReader(new
InputStreamReader(System.in));

        for (int cnt = 0; cnt < 4; cnt++) {
            int num = 0;
            do
            {
                System.out.println("Enter the number of couple (1 - 14) ");
                try
                {
                    num = Integer.parseInt(sc.nextLine());
                    // OR
                    // num = Integer.parseInt(kb.readLine());
                }
                catch (NumberFormatException e)
                {
                    num = 0;
                    System.out.println("Not number");
                }
            } while (num < 1 || num > 14);
            couple[num-1].setInComp();
        }

        String afvString = String.format("%-10s%-25s%-25s", "No.", "Name",
"Contact number");
        System.out.println(afvString);
        int validVotes = 0;
        int invalidVotes = 0;
        /* Needed when working with BufferedReader
        * try {
        * String line = vFile.readLine();
        * while (line != null) { */

        while (sf.hasNext()) {
            String line = sf.nextLine();
            String[] temp1 = line.split(";");
            String[] temp2 = temp1[1].split("#");
            String name = temp1[0];

```

```

    int age = Integer.parseInt(temp2[0]);
    String cell = temp2[1];
    int vote = Integer.parseInt(temp1[2]);
    Voter voter = new Voter(name, cell, age, vote);
    if (voter.getAge() >= 18) {
        boolean found = false;
        for (int k = 0; k < 14; k++)
        {
            if (voter.getVote() == couple[k].getNum() &&
!couple[k].isInComp())
                found = true;
        }
        if (found == false)
        {
            validVotes++;
            System.out.println(validVotes + "\t" + voter);
            couple[vote-1].addVote(); // count individual couple votes
        }
        else {
            invalidVotes++;
        }
    }
    else
    {
        invalidVotes++;
    }
    // line = vFile.readLine();
}

/* needed when working with BufferedReader
}
catch (FileNotFoundException e) {
    System.out.println(e);
}
catch (Exception f) {
    System.out.println(f);
}*/
System.out.println();
System.out.println("Invalid entries: " + invalidVotes);
System.out.println("Valid entries: " + validVotes);
System.out.println("\n");
}

public void displayCouples(){
    System.out.println("\n\nVotes received during week 5");
    System.out.printf(String.format("%-15s%-10s\n", "Couple", "Votes"));
    String output = "";
    for (int cnt = 0; cnt < 14; cnt++) {
        System.out.println(couple[cnt]);
    }
    System.out.println(output);
    System.out.print("Couple(s) who may be eliminated next: ");
    int lowest = getLow();

    for (int cnt = 0; cnt < 14; cnt++) {
        if (lowest == couple[cnt].getNumVotes())
        {
            System.out.print(couple[cnt].getNum() + " ");
        }
    }
    System.out.println("\n");
}
}

```

```

public int getLow() {
    int low = 500;
    for (int cnt = 0; cnt < 14; cnt++) {
        if (couple[cnt].getNumVotes() < low && couple[cnt].getNumVotes() > 0)
        {
            low = couple[cnt].getNumVotes();
        }
    }
    return low;
}
}

```

3.4 TEST/DRIVER CLASS (VotingMenu):

//POSSIBLE SOLUTION FOR QUESTION 3 WITH OOP

```

import java.io.*;
import java.util.Scanner;

public class VotingMenu {

    public static void main(String[] args) throws IOException
    {
        VotingProcess vote = new VotingProcess();
        Scanner sc = new Scanner(System.in);
        // OR
        // BufferedReader kb = new BufferedReader(new
InputStreamReader(System.in));
        char choice = ' ';
        do {
            System.out.println("    MENU\n");
            System.out.println("Option A");
            System.out.println("Option B");
            System.out.println("");
            System.out.println("Q - QUIT");
            System.out.println("\nYour choice? ");

            choice = sc.nextLine().toUpperCase().charAt(0);
            //OR
            //choice = kb.readLine().toUpperCase().charAt(0);
            switch (choice) {
                case 'A':
                    // OPTION A CODE HERE
                    vote.readCouples();
                    vote.readFromFile();
                    break;
                case 'B':
                    // OPTION B CODE HERE
                    vote.displayCouples();
                    break;

                case 'Q':
                    System.out.println("QUIT");
            }
        } while (choice != 'Q');
    }
}

```

ANNEXURE K: SOLUTION without OOP – QUESTION 3: JAVA

```
// A solution for Question 3 WITHOUT using OOP

import java.io.*;
import java.util.Scanner;

public class VotingProcess
{
    int[] dancers = new int[14];
    int[] numVotes = new int[14];
    int[] elim = new int[4];

    Scanner sc = new Scanner(System.in);
    // OR
    // BufferedReader kb = new BufferedReader(new
InputStreamReader(System.in));

    VotingProcess() throws IOException {
        menu();
    }

    public void menu() throws IOException {
        char choice = ' ';
        do {
            System.out.println("    MENU\n");
            System.out.println("Option A");
            System.out.println("Option B");
            System.out.println("");
            System.out.println("Q - QUIT");
            System.out.println("\nYour choice?  ");

            choice = sc.nextLine().toUpperCase().charAt(0);
            // OR
            // choice = kb.readLine().toUpperCase().charAt(0);
            switch (choice) {
                case 'A':
                    // OPTION A CODE HERE
                    readCouples();
                    readFromFile();
                    break;
                case 'B':
                    // OPTION B CODE HERE
                    displayCouples();
                    break;
                case 'Q':
                    System.out.println("QUIT");
            }
        } while (choice != 'Q');
    }

    public void readCouples() {
        for (int cnt = 0; cnt < 14; cnt++) {
            dancers[cnt] = cnt + 1;
            numVotes[cnt] = 0;
        }
    }

    public void readFromFile() throws IOException
    {
        Scanner sf = new Scanner(new FileReader("DataQ3.txt"));
        Scanner sc = new Scanner(System.in);
    }
}
```

```

// OR  BufferedReader vFile = new BufferedReader(new
FileReader("DataQ3.txt"));
// BufferedReader kb = new BufferedReader(new
InputStreamReader(System.in));
for (int cnt = 0; cnt < 4; cnt++) {
    int num = 0;
    do {
        System.out.println("Enter the number of couple (1 - 14) ");
        try
        {
            num = Integer.parseInt(sc.nextLine());
            // OR
            // num = Integer.parseInt(kb.readLine());
        }
        catch (NumberFormatException e)
        {
            num = 0;
            System.out.println("Not number");
        }
    } while (num < 1 || num > 14);
    elim[cnt] = num;
}

String afvString = String.format("%-10s%-25s%-25s", "No.", "Name",
"Cellphone number");
System.out.println(afvString);
int validVotes = 0;
int invalidVotes = 0;
// Needed when working with BufferedReader
// try {
// String line = vFile.readLine();
// while (line != null)
    while (sf.hasNext()) {
        String line = sf.nextLine();
        String[] temp1 = line.split(";");
        String[] temp2 = temp1[1].split("#");
        String name = temp1[0];
        int age = Integer.parseInt(temp2[0]);
        String cell = temp2[1];
        int vote = Integer.parseInt(temp1[2]);
        if (age >= 18) {
            boolean found = false;
            for (int k = 0; k<4; k++)
                if (vote == elim[k])
                    found = true;

            if (found == false) {
                validVotes++;
                if (cell.substring(0,3).equals("+27")){
                    afvString = String.format("%-10s%-25s%-
25s", validVotes, name, cell);
                }
                else
                {
                    afvString = String.format("%-10s%-25s%-25s%-
25s", validVotes, name, cell, "International vote");
                }
                System.out.println(afvString);
                numVotes[vote - 1]++; // count individual couple votes
            } else {
                numVotes[vote - 1] = 0;
                invalidVotes++;
            }
        }
    }

```

```

        }
    } else {
        invalidVotes++;
    }
    //line = vFile.readLine();
}
/* Needed when working with BufferedReader
} catch (FileNotFoundException e) {
    System.out.println(e);
} catch (Exception f) {
    System.out.println(f);
}*/
System.out.println();
System.out.println("Invalid entries: " + invalidVotes);
System.out.println("Valid entries: " + validVotes);
System.out.println("\n");
}
public void displayCouples(){
    System.out.println("\n\nVotes received during week 5");
    System.out.printf(String.format("%-15s%-10s\n", "Couple", "Votes"));
    String output = "";
    for (int cnt = 0; cnt < 14; cnt++) {
        if(numVotes[cnt] > 0)
            { output = output + String.format("%-15s%-10s", dancers[cnt],
numVotes[cnt]) + "\n";
            }
        else
            {
                output = output + String.format("%-15s%-10s",
dancers[cnt], "(Eliminated)") + "\n";
            }
    }
    System.out.println(output);
    System.out.print("Couple(s) who may be eliminated next: ");
    int lowest = getLow();
    for (int cnt = 0; cnt < 14; cnt++) {
        if (lowest == numVotes[cnt]) {
            System.out.print(dancers[cnt] + " ");
        }
    }
    System.out.println("\n");
}

public int getLow() {
    int low = 500;
    for (int cnt = 0; cnt < 14; cnt++) {
        if (numVotes[cnt] < low && numVotes[cnt] > 0) {
            low = numVotes[cnt];
        }
    }
    return low;
}

public static void main(String[] args) throws IOException {
    new VotingProcess();
}
}

```

TOTAL: 120



basic education

Department:
Basic Education
REPUBLIC OF SOUTH AFRICA

**NATIONAL
SENIOR CERTIFICATE**

GRADE 12

INFORMATION TECHNOLOGY P2

NOVEMBER 2013

MARKS: 180

TIME: 3 hours

This question paper consists of 17 pages.

INSTRUCTIONS AND INFORMATION

1. This question paper consists of FIVE sections subdivided as follows:

SECTION A: Multiple-choice questions	(10)
SECTION B: Hardware and software	(49)
SECTION C: Applications and implications	(23)
SECTION D: Programming and software development	(48)
SECTION E: Integrated scenario	(50)
2. Answer ALL the questions.
3. Read ALL the questions carefully.
4. The mark allocation generally gives an indication of the number of facts/reasons required.
5. Number the answers correctly according to the numbering system used in this question paper.
6. Write neatly and legibly.

SECTION A: MULTIPLE-CHOICE QUESTIONS**QUESTION 1**

Various options are given as possible answers to the following questions. Choose the answer and write only the letter (A–D) next to the question number (1.1–1.10) in the ANSWER BOOK.

- 1.1 A ... is the smallest addressable unit of space on a hard disk drive.
- A sector
 - B cluster
 - C track
 - D partition
- (1)
- 1.2 The encryption protocol using public key encryption for encoding data sent over the Internet is known as ...
- A HTML.
 - B FTP.
 - C TCP/IP.
 - D SSL.
- (1)
- 1.3 Which ONE of the following is NOT part of the machine cycle?
- A Fetch data.
 - B Decode instruction.
 - C Encode data.
 - D Execute instruction.
- (1)
- 1.4 A hardware interrupt is ...
- A a method used by peripherals to gain the CPU's attention when they need to communicate.
 - B used to stop a printer when it continues printing an unwanted print job.
 - C used by the memory of a computer to interrupt (or stop) a peripheral when communicating with the CPU.
 - D the process that occurs when the power supply unit fails and the computer stops working.
- (1)
- 1.5 Quad pumping is ...
- A accessing RAM four times per clock cycle.
 - B using the CPU four times per clock cycle.
 - C a pipelining technique.
 - D an alternative to hyperthreading.
- (1)

1.6 Which ONE of the following Boolean expressions is correct?

	EXPRESSION	RESULT
A	TRUE AND NOT FALSE	FALSE
B	FALSE AND (TRUE OR NOT FALSE)	TRUE
C	TRUE OR NOT (TRUE OR NOT FALSE)	TRUE
D	TRUE AND NOT (TRUE AND FALSE)	FALSE

(1)

1.7 A rootkit is ...

- A a backdoor program.
 - B malicious software that activates when a computer system boots up.
 - C a program that logs key presses.
 - D a kit of programs that a hacker uses to infiltrate computer systems.
- (1)

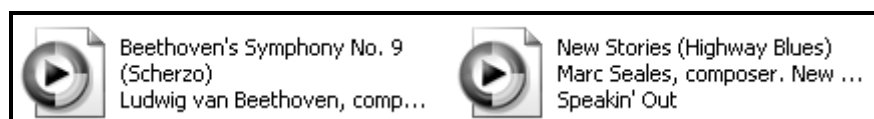
1.8 ... takes place when the sender address and other parts of an e-mail header are altered to appear as though the e-mail originated from a different source.

- A Hoaxing
 - B Phishing
 - C Spoofing
 - D Pharming
- (1)

1.9 Which ONE of the following is an example of a correctly constructed URL for a secure website?

- A <http://www.questionbank.co.za>
 - B <https://www.myonlineshop.co.za>
 - C <http://www.secure.co.za>
 - D <htts://www.secureweb.com>
- (1)

1.10 The two files below are examples of music files.



Which ONE of the following file extensions is most likely to be associated with these two files?

- A wma
 - B docx
 - C bmp
 - D ico
- (1)

TOTAL SECTION A: 10

SCENARIO

A countrywide competition is being held in an effort to discover singing, dancing and acting talent among young people. A website has been created to promote the competition. Each week elimination rounds will be held and judged at different centres around the country. Managers have been appointed at the centres to capture and save results, type reports and ensure that video footage of each performance is captured.

SECTION B: HARDWARE AND SOFTWARE**QUESTION 2**

- 2.1 The managers at the centres were given a choice to use one of the following two devices to perform their tasks:

DEVICE	SPECIFICATIONS
Laptop	- Intel Core i7 2.4 GHz CPU - 8 GB RAM - 1.5 TB HDD
Tablet	- nVidia Tegra 1.2 GHz CPU - 1 GB RAM - 32 GB SSD

- 2.1.1 Give TWO reasons why the managers requested to be issued with the laptop rather than the tablet. (2)
- 2.1.2 A tablet uses an SSD for storage of data and files.
- (a) What does the acronym *SSD* stand for? (1)
- (b) List THREE comparative differences between the features and performance of an SSD and an HDD in **table format**. (3)
- 2.1.3 The tablet has a system recovery partition on its SSD.
- (a) Explain what *partitioning* of a disk means. (2)
- (b) Briefly explain why there is a need for a system recovery partition, specifically on tablets. (1)
- 2.1.4 Both a tablet and a laptop use cache memory. Explain the purpose of cache memory in a computer system. (3)
- 2.1.5 DDR RAM improves the transfer of data between the RAM and the CPU when compared to SD RAM. Explain how this improvement is accomplished. (3)
- 2.1.6 Give TWO reasons why a tablet and a laptop cannot use the same type of RAM. (2)

- 2.2 All the laptop and desktop computers that will be used during the competition have Intel Core i7 CPUs while the tablets have ARM CPUs.
- 2.2.1 List FOUR of the basic components a CPU must contain to be able to execute instructions. (4)
- 2.2.2 Different CPUs use different instruction sets. What is a *CPU instruction set*? (2)
- 2.2.3 Both types of CPU support multithreading. Explain the concept of *multithreading*. (2)
- 2.2.4 Name TWO physical aspects of CPUs that are built in during the manufacturing process to improve the performance of CPUs. (2)
- 2.3 The managers have been told that they can use FireWire 800 or USB 3.0 to upload video material of the performances of contestants to the competition's website.
- 2.3.1 State TWO advantages of USB over FireWire. (2)
- 2.3.2 Would the software used for the editing of videos be considered application software or system software? (1)
- 2.4 Cellular modems are used to connect to the Internet and upload the results of the competition each week. The results are stored in a 'cloud'.
- 2.4.1 Name the type of connection the modem will use with the uploading of results. (1)
- 2.4.2 Explain what is meant by the term *cloud* in this context. (2)
- 2.4.3 Is the storing of results in a cloud a good idea? Motivate your answer. (2)
- 2.5 The results need to be printed each week. The volume of printing is large because of the high number of contestants at each venue.
- 2.5.1 The managers decided to use inkjet printers because they are less expensive. Persuade them to use laser printers instead by giving TWO good arguments in favour of laser printers. (2)
- 2.5.2 Spooling is a technique that is used to accommodate large volumes of printing. Explain what *spooling* is. (2)
- 2.5.3 The USB-port used to connect the printer uses serial data transfer. Explain the difference between *serial* and *parallel* data transfer. (2)

- 2.6 Each week the managers have a group conference call to discuss results and some of the outstanding performances. It is suggested that they use VoIP in order to save costs.
- 2.6.1 Briefly explain the basic concept *VoIP*. (2)
- 2.6.2 Name a VoIP client that the managers could use to host the conference call. (1)
- 2.6.3 Name ONE other way (besides a conference call) in which the managers could have discussions and collaborate online. (1)
- 2.7 The error message 'Missing DLL files' appeared when one of the managers wanted to use an application installed on his laptop.
- 2.7.1 Give ONE possible reason why the DLL files are missing. (1)
- 2.7.2 What needs to be done so that the manager will be able to use the software again? (1)
- 2.7.3 When new software is installed on a computer system running a Windows operating system an entry is made into the registry of the system. Explain in computing terms what the *registry* is. (2)
- TOTAL SECTION B: 49**

SECTION C: APPLICATIONS AND IMPLICATIONS**QUESTION 3: e-COMMUNICATION**

Friends and family of the contestants requested that video footage of the two best performances after each round be made available to them to watch again.

- 3.1 Someone requested the managers to e-mail them the video footage as an attachment. Give TWO reasons why this request is possibly not feasible. (2)
- 3.2 The organisers made the videos available as podcasts.
- 3.2.1 Explain the concept of *podcasting*. (1)
- 3.2.2 Why will podcasting be an advantage for people who want to watch the video more than once? (1)
- 3.2.3 Name ONE example of an electronic medium that can be used to inform friends and family when a new podcast is available. (1)
- 3.3 Friends and family who want to download multimedia files of all the events must preferably have a broadband Internet connection.
- 3.3.1 Explain the term *broadband*. (2)
- 3.3.2 ADSL is an example of a cabled broadband connection.
- (a) Is ADSL a digital or an analogue connection? (1)
- (b) Give ONE advantage of an ADSL connection. (1)
- 3.3.3 Give an example of another cabled broadband connection. (1)
- [10]**

QUESTION 4: SOCIAL AND ETHICAL ISSUES

The organisers are hosting a website for the competition that publishes results and information about the contestants.

- 4.1 The parents of the contestants are concerned about personal information such as contact details that are published on the competition's website.
- 4.1.1 Justify the parents' concerns by describing TWO ways in which personal information provided on a website can be misused. (2)
- 4.1.2 Suggest ONE measure that a contestant can take to protect personal information that needs to be published on the Internet. (1)
- 4.2 Even though personal information can be protected an activity known as social engineering could be used to gain access to the information.
- 4.2.1 Explain what *social engineering* is. (2)
- 4.2.2 Describe ONE way in which social engineering can be prevented. (1)
- 4.3 One of the contestants has been removed from the competition because he tweeted insulting remarks about another contestant.
- 4.3.1 What is meant by the term *tweet*? (1)
- 4.3.2 What term is used to refer to the practice of publishing insulting remarks about people on an online or public forum? (1)
- 4.4 Some of the contestants want the organisers to create a blog for the competition.
- 4.4.1 What is a *blog*? (1)
- 4.4.2 How would the contestants be advantaged by the blog? (1)
- 4.5 Entries must be done electronically on the website. A request was made that there must also be an alternative way to handle the entries because of the digital divide in South Africa. What is meant by the term *digital divide*? (1)
- 4.6 Members of the IT team can work on the voting system from home instead of at the head office. Discuss the disadvantages of working from home from the IT members' point of view. (2)

[13]**TOTAL SECTION C: 23**

SECTION D: PROGRAMMING AND SOFTWARE DEVELOPMENT**QUESTION 5: ALGORITHMS AND PLANNING**

The organisers of the competition need to capture, store and manage information about the contestants. An IT team was appointed to develop software to help the organisers to perform these tasks.

5.1 Before the start of the competition the IT team had to create a user-friendly interface for the organisers to capture information about the contestants. They had to decide on suitable controls for the GUI for the entering of the information.

5.1.1 The category options of the competition are Dance, Drama and Music. The following two controls have been suggested:

Suggestion A:



A screenshot of a GUI control labeled 'Category:'. It features a dropdown menu with a small downward arrow on the right side. The menu is open, showing three options: 'Dance', 'Drama', and 'Music' listed vertically.

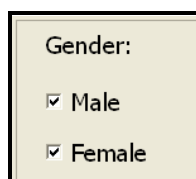
Suggestion B:



A screenshot of a GUI control labeled 'Category:'. It consists of a simple rectangular text input field with a thin border.

Give TWO reasons why Suggestion A is a better choice. (2)

5.1.2 The input control in the screen shot below may cause users to select both options when they enter the gender of a contestant.

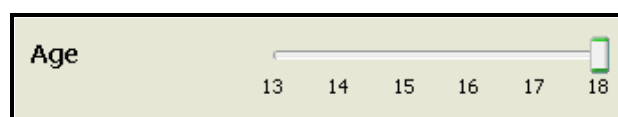


A screenshot of a GUI control labeled 'Gender:'. It contains two checkboxes, one for 'Male' and one for 'Female'. Both checkboxes have a checkmark in the box to their left, indicating they are both selected.

Suggest how this situation can be prevented. (1)

5.1.3 Only contestants between the ages of 13 and 18 are allowed to take part in the competition.

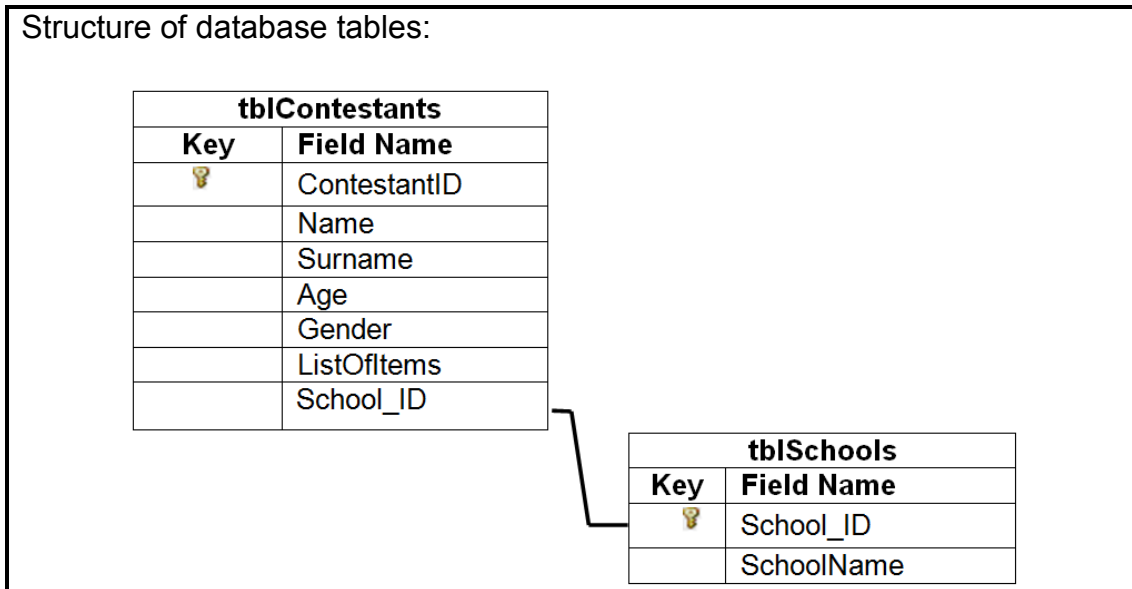
The following control is used to enter the age of the contestants:



A screenshot of a GUI control labeled 'Age'. It is a range slider with a horizontal track. Below the track, the numbers 13, 14, 15, 16, 17, and 18 are marked. A green vertical bar is positioned at the right end of the track, corresponding to the value 18.

Explain why data entered using this control will be valid but not necessarily correct. Use an example with actual values as part of your explanation. (2)

5.2 A database has been created with two tables (**tblContestants** and **tblSchools**) containing data about the contestants. Study the structure of the database tables along with the additional information below and answer the questions that follow.



Additional information:

- Contestants may enter for up to three items.
- Each item is performed in a specific category.
- The **ListOfItems** field contains the list of items the contestant entered for in the following format:

`<Category, Description of first item>;<Category, Description of second item>;<Category, Description of third item>`

EXAMPLE: Joanne Johnson is registered to perform a monologue in the drama category and a keyboard item in the music category. For this contestant the **ListOfItems** field in the **tblContestants** table contains the following information:

Drama, Monologue; Music, Keyboard

5.2.1 When the secretary tries to add data of newly registered contestants into the **tblContestants** table, an error message is displayed when she enters a value into the **School_ID** field.

- (a) Select the type of error that occurs in this case from the given list below. Write only the letter (A–C) next to the question number (5.2.1(a)).

- A Update anomaly
- B Insert anomaly
- C Delete anomaly

(1)

- (b) What needs to be done to be able to register the new contestants that cannot be registered currently? (1)

5.2.2 It has been decided to use a database rather than a text file to store the data about the contestants.

- (a) Give TWO reasons why a database is a better structure to keep data about contestants than a text file. (2)

- (b) What type of key is the **School_ID** field in the **tblContestants** table? (1)

- (c) Another table needs to be added to convert the database into first normal form. Draw the new table, indicating the name of the table and appropriate fields and keys.

NOTE: Draw only the new table and NOT the two given tables. (5)

- (d) Explain the relationship between the newly defined table in QUESTION 5.2.2(c) and the relevant given table (**tblContestants** or **tblSchools**) with specific reference to the fields involved. (2)

5.2.3 When developing a program to manipulate the database it was decided to work with objects.

- (a) Define an *object* in computer programming terms. (2)

- (b) Give ONE example of an object that needs to be created for this program. (1)

- (c) A class for this program has been defined with public attributes. Give TWO reasons why this is NOT good programming practice. (2)

5.3 The name and score of each contestant in the Drama category of the competition are saved using two parallel arrays named **arrNames** and **arrScores**. The scores are in the range 0–100 (both included).

The following algorithm was developed to determine the winner:

Line	
1	highest score \leftarrow 100
2	Initialise loop counter
3	Repeat for all the elements of arrScores
4	if highest score < arrScores [counter]
5	arrScores [counter] \leftarrow highest score
6	End if
7	Increase loop counter
8	End repeat
9	Display highest score

5.3.1 The output will not be correct due to TWO logical errors. Identify the instructions causing the logical errors and write down their line numbers and the correct instructions. (4)

5.3.2 The **name** of the contestant with the highest score must also be displayed.

Write line(s) of pseudocode that need to be added to the given algorithm to accomplish this task. Use the line numbers to indicate the position where the new line(s) of pseudocode should be inserted into the given algorithm. (4)

5.4 Programmers who created the software were requested to create a test plan along with other technical documentation to ensure that the software is up to standard.

5.4.1 The system design document (SDD) is one of the first documents developed. List THREE aspects that form part of the SDD. (3)

5.4.2 List THREE aspects that should be included in the test plan. (3)

5.4.3 Why is it important to include hardware and software requirements in user documentation? (1)

5.5 A text file is used to store the date, time and password of each user that logs onto the local server at each centre where the elimination rounds of the competition take place.

Example of the content of the text file:

```
Date: 15/08/2013
Time: 15:00:00
Password: AB1002
Date: 15/08/2013
Time: 15:30:00
Password: PS1022
:
```

Activities are suspicious when either of the following applies:

- A person logs onto the network before 07:30 in the morning or after 16:00 in the afternoon.
- The password is not in the prescribed format – two capital letters and four digits.

An algorithm is supplied that should be able to read the data from the text file, to test whether the time or password is suspicious (meets the conditions set out above) and to write all the information of a suspicious activity to a new text file.

(The algorithm is given on the next page.)

Line	
1	Open text file with logon information
2	Open new suspicious_activity file
3	Repeat while not end of log file
4	Read three lines from log file
5	Extract the date, time and password
6	validTime ← call code to test for a valid login time with the hour and minute when the user logged in as parameters
7	validPass ← call code to test for a valid password with the password as parameter
8	IF ...
9	Save date, time and password to suspicious_activity file
10	Close the log file
11	End repeat
12	Close suspicious_activity file

5.5.1 The function/method called in line 6 uses the following two parameters:

- Hour
- Minute

(a) What are *parameters* in terms of programming? (2)

(b) Suggest a suitable data type for these parameters. Explain your choice of data type. (2)

5.5.2 The function/method called in line 7 will use 'try ...' protection blocks.

Write down the general structure of a 'try ...' protection block AND briefly explain the purpose of each section. (3)

5.5.3 Write down the logical condition to complete the IF-statement in line 8 of the given algorithm. (2)

5.5.4 If this algorithm is correctly coded and executed, it will cause an error in line 4 due to a mistake that was made in line 10.

(a) Explain the cause of the error. (1)

(b) Briefly explain how to correct the mistake. (1)

TOTAL SECTION D: 48

SECTION E: INTEGRATED SCENARIO**QUESTION 6**

During the final round of the competition the audience in the auditorium votes for contestants to determine a winner. The event is also broadcast live on television. Members of the television audience at home are also able to vote by SMS.

- 6.1 The audience in the auditorium has been provided with remote voting controllers that allow them to press a button to transmit their vote wirelessly via Bluetooth technology to a local server at the auditorium.
- 6.1.1 Is Bluetooth a suitable technology to use in this scenario? Motivate your answer. (2)
- 6.1.2 What is the transmission rate of standard Bluetooth devices (3 Mbps or 3 Gbps)? (1)
- 6.1.3 An audience member's voting controller does not connect to the wireless network although the member is within range of the server.
- Name TWO possible troubleshooting steps that can be followed to solve the problem. Replacing the voting controller is not an option. (2)
- 6.2 All the votes received from the audience in the auditorium and SMSs from the television audience are stored in a relational database. The database is maintained by a database administrator.
- 6.2.1 Explain what a *relational database* is. (2)
- 6.2.2 Give the name of ONE example of an existing database system that could be used to create this database. (1)
- 6.2.3 Name TWO specific responsibilities of a database administrator. (2)
- 6.3 The live broadcast of the final round of the event can be streamed.
- 6.3.1 Explain what is meant by *streamed*. (2)
- 6.3.2 Name TWO components that are required to be able to stream a live event. (2)
- 6.3.3 An entrepreneur is recording the live broadcast on his home computer. He plans to sell copies of the recording. Discuss the ethical issues of these activities. (2)

- 6.4 An existing cabled network is used by the judges at the event centre to transmit their scores electronically to a server at the centre. The organisers are planning to use a wireless network for this purpose in future.
- 6.4.1 State TWO advantages of a wireless network over a cabled network. (2)
- 6.4.2 State TWO additional hardware components that are required to create the wireless network. (2)
- 6.4.3 Which network protocol would be appropriate to use for the wireless network? (1)
- 6.5 While uploading results, one of the organisers noticed suspicious activities on his computer. He suspects unauthorised software is running on his computer to gain access to competition results.
- 6.5.1 State TWO types of programs that can be used to retrieve data from a computer without the permission or knowledge of the user. (2)
- 6.5.2 Explain how the organiser could check whether there is any unauthorised software running on his computer. (1)
- 6.5.3 Describe TWO ways in which the organisers can prevent their computers from being accessed by unauthorised software. (2)
- 6.6 Some judges use Apple Mac computers while others use computers running on the Microsoft Windows operating system. Both these operating systems are proprietary software.
- 6.6.1 Describe THREE functions of an operating system. (3)
- 6.6.2 State TWO advantages of a proprietary operating system over an open source operating system. (2)
- 6.6.3 Explain why software designed for the Apple Mac cannot run on a computer that uses the Microsoft Windows operating system. (2)
- 6.7 The organisers of the competition have been advised to use cloud storage for the video material.
- 6.7.1 The cloud server probably uses RAID technology. Briefly explain what *RAID* is. (Expanding the acronym will NOT be accepted as an answer.) (2)

- 6.7.2 Several levels of RAID can be implemented. Motivate why the cloud server will probably use RAID 1 rather than RAID 0. (2)
- 6.7.3 'Cloud computing' can be beneficial for green solutions in terms of hardware. Explain this statement briefly. (2)
- 6.8 Projects such as this create a variety of IT-related job opportunities.
- 6.8.1 Give a summary of the main responsibilities of a systems analyst. (2)
- 6.8.2 Apart from a database administrator and a systems analyst, name TWO other types of IT specialists who will most probably be employed by the organiser of this project. (2)
- 6.9 ICT provides more effective solutions for the capturing of data and generating results than a paper-based voting system.
- Motivate why the above statement is valid by referring to TWO of the following aspects:
- Faster and improved data processing
 - Improved data storage
 - Improved communication (2)
- 6.10 Some of the parents who do not use the Internet regularly often need some basic information about browsing the Internet.
- 6.10.1 Give ONE example of a well-known Internet search engine. (1)
- 6.10.2 Give TWO hints for conducting an effective Internet search AND give an example with each hint which is applicable to a search for the website for this competition. (4)
- TOTAL SECTION E: 50**
GRAND TOTAL: 180



basic education

Department:
Basic Education
REPUBLIC OF SOUTH AFRICA

**NATIONAL
SENIOR CERTIFICATE**

GRADE 12

INFORMATION TECHNOLOGY P2

NOVEMBER 2013

MEMORANDUM

MARKS: 180

This memorandum consists of 22 pages.

SECTION A: MULTIPLE-CHOICE QUESTIONS**QUESTION 1**

- | | | |
|------|----------|-----|
| 1.1 | A OR B ✓ | (1) |
| 1.2 | D ✓ | (1) |
| 1.3 | C ✓ | (1) |
| 1.4 | A ✓ | (1) |
| 1.5 | A ✓ | (1) |
| 1.6 | C ✓ | (1) |
| 1.7 | B OR D ✓ | (1) |
| 1.8 | B OR C ✓ | (1) |
| 1.9 | B ✓ | (1) |
| 1.10 | A ✓ | (1) |

TOTAL SECTION A: 10

SECTION B: HARDWARE AND SOFTWARE

QUESTION 2

2.1 2.1.1

Any TWO of ✓✓:

The laptop would be preferred because of the following reasons:

- *More powerful CPU/Processing is done faster/Higher processing speed/2.4 GHz CPU/Any valid reason that implies a more powerful CPU*
- *More RAM/8GB RAM/Any valid reason that implies more RAM*
- *Bigger HDD/1.5TB HDD/It has more storage space to save all the video footage*
- *Need a keyboard to do a significant amount of typing/Any valid reason that implies the need of a keyboard*
- *A larger screen to work on*

Also accept any other correct sensible reason

NOTE: • If learner only answered *Faster/Better* may only receive ONE MARK but any other answer may not imply faster.

• **DO NOT** accept any answer about software. (2)

2.1.2 (a) **SSD: Solid State Drive/Disc OR Static State Drive**✓ (1)

(b) *Any THREE comparative differences* ✓✓✓

SSD	HDD
<i>Both sides must be correct to earn one mark:</i>	
• Electronic	• Magnetic
<i>For the following only the left/right half is required:</i>	
• Has no moving parts	• Has moving such as platters and read/write head
• Faster data transfer rate	• Slower data transfer rate
• Smaller storage capacity	• Larger storage capacity
• Smaller (physical dimensions)	• Larger (physical dimensions)
• More robust/reliable	• Less robust/reliable
• Less power consumption/more energy efficient	• More power consumption/Less energy efficient

NOTE: • No half marks
• Do not accept cost as an answer (3)

2.1.3 (a) A (single) physical hard drive✓ is (logically/virtually) divided into logical drives/(separate parts)/partitions✓ (2)

(b) *Any ONE sensible reason for having a system recovery partition:✓*

- Recover the operating system
- So that the operating system (OS) can easily be restored to factory settings.
- It is difficult to reinstall an operating system on a tablet. The system recovery partition contains a backup of the operating system – makes is easy to recovery the OS.

Also accept any correct alternative reason (1)

2.1.4 *Any ONE of:*

- Cache memory is high speed/faster memory✓ that stores instructions and data that will probably be used next✓ by the CPU✓.

OR

- Cache memory increases the speed of the system✓ by using its a high speed memory that stores instructions and data that will probably be used next✓ by the CPU✓.

OR

- Cache memory increases the speed of the system✓ by preventing the CPU having to wait✓ for data from the slower RAM✓

NOTE: Reference to alleviating the bottleneck – one mark (3)

2.1.5 Use quad pumping/access occur four times (double pumping/access occur two times) during each clock cycle with quad/double pumping. Access occurs during the stable low, the change of voltage (low to high and again high to low) and the high voltage stage ✓✓✓

OR

Multiple✓ access✓ per cycle✓

OR

Quad/double pumping✓

Also accept any correct alternative explanation of double/quad pumping. (3)

- 2.1.6 Any *TWO* reasons ✓✓:
- The **RAM in a tablet** is not removable (fixed) from the motherboard/Not modular
 - **RAM in a laptop** is upgradeable/**RAM in a tablet** cannot be upgraded
 - Motherboards differ
 - Physical space for RAM is different/The RAM in a tablet is specialised memory designed to be as small as possible and is unique to that device as it is physically a part of the logic board
 - RAM in a laptop is generic and can be used in different types of laptops (2)
- 2.2 2.2.1 Any *FOUR* of: ✓✓✓✓
- Registers
 - Arithmetic Logic Unit (ALU)
 - Floating Point Unit (FPU)
 - Control Unit (CU)
 - Cache (4)
- 2.2.2 The *instruction set* is a (unique) set of instructions/rules✓ that the processor is able to execute✓.
- OR**
- The *instruction set* is a collection of machine code instructions which enable a processor to carry out all its tasks. (2)
- 2.2.3 *Multithreading* allows processes to be split into "threads" ✓ which can run simultaneously through the CPU✓.
- Also accept any correct alternative explanation using a correct example.* (2)
- 2.2.4 Any *TWO* of:✓✓
- Increase number of transistors on CPU/Decrease the size of the transistors
 - Increase clock speed of CPU
 - Increase the amount of cache memory
 - Choice of instruction set/Add additional instruction sets
 - Increase the number of pipelines
 - Make use of multiple cores
 - Increase the size of the registers
 - Hyper-threading
- NOTE:** No need to refer to the increase/decrease (2)

- 2.3 2.3.1 Any *TWO* advantages of USB over Firewire: ✓✓
- USB is faster than Firewire.
 - USB can daisy chain more devices than Firewire
 - USB is more common/better supported/more devices make use of USB
 - USB is more cost effective than using Firewire (2)
- 2.3.2 Application software ✓ (1)
- 2.4 2.4.1 Any *ONE* of: ✓
- 3G
 - EDGE
 - GPRS
 - LTE/4G
 - Radio
- NOTE:** Do not accept WiFi; Wireless (1)
- 2.4.2 The *cloud* refers to storage/services/space✓ located on the Internet/on-line/remotely/external✓ (2)
- 2.4.3 Yes/+ OR No/-
- NOTE:**
- If only Yes/No – no marks
 - Mark allocation:
 - ONE mark for taking a stand/Yes/No/+/- with a motivation irrespective if correct ✓
 - ONE mark for valid/correct motivation ✓
- If answered YES/positive/confirm:**
- Data will be automatically backed up/saved somewhere.
 - The data will be available from anywhere via the Internet/remote computer.
 - No need to maintain/manage your own storage media.
- If answered NO/negative:**
- Internet connection is required to access data.
 - Can be slow to access due to slow Internet connection.
 - You have no control over the location where the files are saved.
 - Security of sensitive data/personal information contained in documents may be a concern. (2)

- 2.5 2.5.1 *Any TWO good arguments in favour of laser printers: ✓✓*
- Reduce operational costs (toner/ink cartridges).
 - Have greater page per minute rates than inkjet printers/Faster.
 - Printing done is not water soluble/does not get smudged/is more durable.
 - Better print quality
- Also accept any other correct alternative argument* (2)
- 2.5.2 *Spooling is the process of buffering (temporary storing of)✓ content while waiting for slower printer to become available ✓* (2)
- 2.5.3 *Parallel data transfer means the number of bits can be transferred at the same time along many connections✓ while serial data transfer means the bits are transferred one after the other along a single connection.✓* (2)
- 2.6 2.6.1 VoIP converts (analogue) voice/video/images data into (digital) data✓ that can be sent over a network/Internet✓
OR
 VoIP can use a standard Internet connection✓ to make telephone calls✓
- NOTE:** VoIP is voice over Internet protocol – ONLY ONE MARK (2)
- 2.6.2 *Any ONE of: ✓*
- Skype
 - Google Hangouts/Google Talk
- Also accept any other correct VoIP client* (1)
- 2.6.3 *Any ONE of:✓*
- Discussion groups such as chat rooms, newsgroups, forums
 - IM messaging
 - Social networking sites and blogs such as Facebook, Twitter, MySpace, FaceTime, IRC etc.
- Also accept any other appropriate online collaboration tool* (1)
- 2.7 2.7.1 *Any ONE of: ✓*
- The application/file was deleted manually/The application was not uninstalled correctly
 - Not installed properly/fully
 - The DLL files got corrupted by malicious software/hardware failure.
 - Hard drive/file corruption
 - Incorrect shutdown procedures
- Also accept any other correct alternative reason* (1)

- 2.7.2 Any ONE of: ✓
- The application software needs to be re-installed
 - Repair the registry (on a Windows OS machines)
 - Download/Replace the missing DLL-files
- (1)

- 2.7.3 The *registry* is a database ✓ storing the information about the configuration data of the software/hardware/users program settings, etc. ✓
- (2)

TOTAL SECTION B: 49

SECTION C: APPLICATIONS AND IMPLICATIONS**QUESTION 3: e-COMMUNICATION**

3.1 *Any TWO reasons for not being feasible: ✓✓*

- The video footage is contained in large data files
- Netiquette requires not sending large attachments/Do not e-mail large files
- Large e-mail attachments can be blocked/flagged as spam
- Bandwidth constraints/slow to upload or download
- Data cost per MB/GB

NOTE: DO NOT ACCEPT:

- Any answer on privacy/piracy/format of the video (2)

3.2 3.2.1 *Podcasting* is when video/audio clips are made available/published online for downloading purposes✓. The format caters for portable media devices. (1)

3.2.2 *Any ONE of: ✓*

- Podcasts need to be downloaded only once
- Podcasts can normally be downloaded to a device
- You do not need to be online to watch it after you have downloaded it. (1)

3.2.3 *Any ONE of: ✓*

- RSS feeds
- SMS
- Blog post
- Website update
- e-mail
- instant message (IM)

Also accept any other correct alternative medium (1)

3.3 3.3.1 *Broadband* is any high speed/high bandwidth✓ connection✓ to the Internet. (2)

3.3.2 (a) Digital ✓ (1)

(b) *Any ONE advantage of ADSL connection ✓:*

- permanent connection
- you can be browsing the web and receive/make telephone calls at the same time
- cheaper than cellular connections
- fixed monthly rate
- stable connection
- faster (1)

3.3.3 *Any ONE broadband alternative to ADSL ✓:*

- ISDN
- Diginet
- Fibre/FTTH/FTTx (1)

[10]

QUESTION 4: SOCIAL AND ETHICAL ISSUES

- 4.1 4.1.1 *Any TWO ways of misusing personal information: ✓✓*
- There may be predators online that could use that information for harm
 - The location of the children may be tracked down
 - Details may be used for identity theft
 - Online fraud
 - Cyber bullying
 - Details may be used to receive spam
- Also accept any other correct sensible answer* (2)
- 4.1.2 *Any ONE way of protecting personal information: ✓*
- Require username and password for contestants only/Create a login for contestants
 - Do not display personal data such as phone numbers and addresses on publicly accessible web pages.
 - Encrypt data/secure site
- Also accept any correct alternative answer* (1)
- 4.2 4.2.1 Social engineering is the human/electronic interaction of an unauthorised person convincing/conning/deceiving✓ an authorised person to give them access✓ to information.
- Also accept any other correct and relevant explanation.* (2)
- 4.2.2 *Any ONE way to prevent social engineering: ✓*
- The training of staff to recognise threats.
 - Strict security measures to prevent contact with unauthorised people.
- Also accept any other correct and relevant explanation* (1)
- 4.3 4.3.1 *Any ONE of ✓:*
- *Tweet* is the use of a social media website called Twitter to send messages
 - Short message
- (1)
- 4.3.2 *Any ONE of: ✓*
- Flaming
 - Trolling
 - Smashing/Smearing
 - Cyber bullying
 - Defamation of character
- (1)

- 4.4 4.4.1 A blog is an online public diary (focused on a specific topic) ✓
Also accept any other correct and relevant explanation (1)
- 4.4.2 *Any ONE advantage of the blog: ✓*
• Some contestants may try to influence public opinion about him/herself
• The contestants may obtain some help from specialised persons in the industry to gain an unfair advantage.
Also accept any other correct and relevant explanation (1)
- 4.5 Digital divide means not everybody has skills/access to computers/Internet and is disadvantaged because of this lack of access. ✓ (1)
- 4.6 *Any TWO of: ✓✓*
• Not enough face-to-face communication with team members and management about the system - can lead to misunderstandings
• No working hours can lead to too much or too little work done.
• Lack of socialisation with other people at work.
• Feel isolated and not part of the team.
• Easily distracted from work
• Lack of technology
Also accept any other relevant and sensible answer (2)
- TOTAL SECTION C: 23**

SECTION D: PROGRAMMING AND SOFTWARE DEVELOPMENT**QUESTION 5: ALGORITHMS AND PLANNING**

- 5.1 5.1.1 Any *TWO* reasons why *combobox* is better choice than *textfield/editbox*: ✓✓
- Only the given options are available to choose from/ Categories supplied/User friendly
 - No spelling mistakes can be made
 - No need for extra validation required as options/answers are limited
 - No typing required – save time (2)
- 5.1.2 Any *ONE* way of preventing both genders selected ✓:
- Rather make use of radio buttons/radiogroup buttons instead of checkboxes.
 - Write code to prevent selection of both checkbox
 - Change selection component so that only one option can be selected for example a combo box or listbox (1)
- 5.1.3 Data will be *valid* because it will be in the correct range ✓ (13 to 18)
It will not necessarily be *correct* because an age of 15 can be given to a contestant of age 18. ✓ (2)
- 5.2 5.2.1 (a) B (Insert anomaly) ✓ (1)
- (b) Add the schools information to the **tblSchools** table before entering the information of the contestants ✓
OR
You can register a learner without entering the School_ID field ✓ (1)
- 5.2.2 (a) Any *TWO* reasons why *database* is better data structure ✓✓
- Referential integrity can be applied to prevent invalid entries
 - Neat reports can be created fairly quickly without requiring programming code
 - Queries can be created and executed on the content of a database
 - The availability of database forms for data input and validation before data is saved
 - Better organisation of data
 - Easier to manage database
 - Allows for different data types to be used
 - Database can be secured with use of passwords (2)
- (b) Foreign key ✓ (1)

tblPerformance		Criteria for the new table	Mark
Key	FieldName	<i>Descriptive table name</i>	✓
(PK)	PerformanceID	<i>Primary Key (a new key) OR use a combined primary key</i>	✓
(FK)	ContestantID	<i>Foreign Key (ContestantID)</i>	✓
	Category	<i>Extra fields splitting the Category and Description</i>	✓✓
	Description		

- (d) *Mark allocation:*
- Reference for one-to-many relationship - ONE mark
 - Linking the correct primary key to correct foreign key – ONE mark

Any ONE of: ✓✓

There is a one-to-many relationship between the tblContestants table (one) and the tblPerformances table (many)

OR

The ContestantID field (primary key) in the tblContestants table will be linked to the ContestantID field (foreign key) in the tblPerformances table creating a one-to-many relationship

NOTE:

- The answer must state which table is on the "one" side and which table is on the "many" side of the relationship.
- The use of a diagram to indicate the newly created relationship is acceptable

- 5.2.3 (a) An object is an instantiated ✓ class definition ✓
OR
An object is an instance of a class (2)

- (b) *Any ONE example of an appropriate object for this scenario ✓*
- Contestant
 - School
 - Performance
 - Any windows GUI control (e.g. button/label, etc.)

NOTE: Also accept object declarations using programming language (even if the syntax is incorrect) (1)

- (c) The declaration allows attributes to be (directly) accessed ✓ from outside the class ✓

NOTE: *Also accept any of the following explanations:*

- Defies OOP encapsulation principles/destroys the principle of hiding/protecting data
- If attributes are public – there is no programming control of access to the data values of the object (2)

- 5.3 5.3.1 **Line number** **Correction**
- 1 ✓ highest score ← 0 ✓
- 5 ✓ highest score ← arrScores[counter] ✓ (4)
- OR**
- The algorithm does not display/keep track of the name of the winner ✓✓
- OR**
- The coding to assign the name of the winner inside the IF-statement (lines 4–6) is missing ✓✓
- Winner ← arrNames[Counter]**
- 5.3.2 *Mark allocation:* (2x2)
- One mark for line number ✓ and one mark for pseudocode ✓
- NOTE:** Line numbers might differ according to learner solution
Insert inside IF-statement (lines 4–6)/ (before or after line 5):
- Name of winner ← arrNames[Counter]**
- Change line 9/10 to:
- Display the name of winner** (4)
- 5.4 5.4.1 *Any THREE DESIGN aspects of the system design document:*
✓✓✓
- A problem statement
 - Did research on input/output
 - The functions and properties of the software
 - Any restrictions/limitation
 - Type of interfaces that will be used, e.g. GUI or text
 - Requirements for data input/output
 - The amount of data the system will need to process
 - Data structures that will be used
 - Requirements regarding security and privacy
 - Any log in requirements
- Also accept any other relevant and sensible answer in respect of designing a software project* (3)

5.4.2 Any **THREE** aspects of a **TEST PLAN** ✓✓✓

- A control list of the programs specifications – make sure the program does what it is supposed to be doing
- Installation test – make sure the program installs correctly on the stated platforms, i.e. Windows/Mac/Linux
- Test input data: Make sure the program handles all data input/validation of data correctly
- Procedure to check the storing of data/the import and export of data
- Procedure to check requirements of printing against printing the system delivers
- Procedure to make sure the program works on various hardware and software configurations

Also accept any other relevant and sensible answer in respect of testing the software project (3)

5.4.3 Hardware and software requirements are required so that prospective clients may view the requirements before buying the software. ✓

Also accept any alternative correct motivation (1)

5.5 5.5.1 (a) *Parameters* are variables/constants/values ✓ that **are passed** to the function/procedure/method to process. ✓ (2)

- (b) *Data type*: Integer/number ✓ (accept all variations of Integer)
Calculations have to done/Do not have to convert string into numbers to do the calculations/Easier to do calculations directly with numbers/Makes provision for the expected range of values ✓

OR

Data type: String ✓
String values can be converted in the coding of the function/method. ✓

NOTE: Do not accept: Real or variations (2)

Structure: DELPHI	JAVA
TRY EXCEPT/FINALLY END	try catch final {optional section}

Explanation of each part:

- Try → code to be executed with possible errors occurring
- Except/Catch → when an error is encountered this code will be executed/handling the exceptions
- Finally/Final → the code is executed irrespective of the error occurring

Mark allocation:

- ONE mark for structure
- TWO marks for explanation of any TWO sections. (3)

5.5.3 NOT(validTime) OR NOT(validPass)
OR
(validTime = False) OR (validPass = False)
OR
NOT ((validTime = True) AND (validPass=True))
OR
NOT (validTime AND validPass)

TWO marks✓✓ or nothing

NOTE:

- Do not penalise for syntax
- Accept any other CORRECT logical condition (2)

5.5.4 (a) The loop tries to read data from the log text file which is closed inside the loop. ✓ (1)

(b) *Correction* ✓

- Move the "close log file" (line 10) to after the end of the while loop, i.e. after line 11.

OR

- Swap line 10 and line 11 (1)

TOTAL SECTION D: 48

SECTION E:**QUESTION 6: INTEGRATED SCENARIO**

- 6.1 6.1.1 **NOTE:**
- Any ONE correct motivation ✓✓
- If answered Yes:**
- Bluetooth has a range of 100 m if signal amplifiers are used.
 - Line of sight is not required
- If answered No:**
- Bluetooth has a limited range (around 10 metres) (the votes of many audience members will not be received by the server) (2)
- 6.1.2 3 Mbps ✓ (1)
- 6.1.3 *Any TWO of ✓✓:*
- The device needs to be paired with the server
 - The controller is not discoverable to the server
 - A password may need to be supplied to pair the two devices
 - The batteries need to be replaced
 - The device/server must be switched on
 - Reset the device/server
 - Bluetooth needs to be activated on both pairing devices (2)
- 6.2 6.2.1 *A relational database is a database with multiple tables✓ with relationships defined between tables✓ using keys. (2)*
- 6.2.2 *Any ONE of: ✓*
- MySQL
 - Microsoft SQL Server (MS SQL)
 - Postgres
 - Oracle
 - Microsoft Access
 - Paradox
- Also accept any other correct relational database software (1)*
- 6.2.3 *Any TWO responsibilities of database administrator: ✓✓*
- Make backups of databases
 - Ensure integrity of data
 - Do queries on data from database
 - Create and maintain user profiles
- Also accept any other correct answers (2)*

- 6.3 6.3.1 *Any ONE of (streamed):*
- Watching a video online✓ while it downloads in the background✓
- OR**
- Media content/video delivered continuously✓ over the Internet✓
- OR**
- Streaming is using a high bandwidth connection✓ to make continuous flow of data possible✓
- OR**
- Transmitted in continuous stream while earlier parts are being used, e.g. watching a live game while it is being played. (2)
- 6.3.2
- (Broadband) Internet connection✓
 - Digital video camera/computer/devices✓ (2)
- 6.3.3 *Any discussion on ONE issue (issue & motivation) OR TWO valid reasons: ✓✓*
- It's illegal to sell pirated copies
 - Bootlegging
 - The person doesn't have permission/rights to download the video from the Internet and then sell the video for personal gain
- Also accept any other correct answers* (2)
- 6.4 6.4.1 *Any TWO advantages of a wireless network over a cabled network: ✓✓*
- Less expensive to install/easy to setup
 - No need to use cables
 - More convenient as clients do not need to be at a network point to access the network
 - Easy to add another access point/computer/device
 - Mobility of devices (2)
- 6.4.2
- Wireless access point/Wireless router with access point ✓
 - WiFi adapter/WiFi network card✓ (2)
- 6.4.3 WiMAX✓
- NOTE: Do not accept: WiFi** (1)

- 6.5 6.5.1 *Any TWO of: ✓✓*
- Trojan horse
 - Key logger
 - Malware
 - Spyware
 - Rootkit
- NOTE:** Do not accept virus/worm (2)
- 6.5.2 *Scan the computer with any ONE of:✓*
- anti-virus
 - anti-malware software
 - anti-spyware
 - using task manager
 - (using the log of the) firewall (1)
- 6.5.3 *Any TWO of ✓✓:*
- (Have an up to date) anti-virus software installed
 - (Have an up to date) anti-spyware software installed
 - (Have an up to date) anti-malware software installed
 - Do not open any suspicious attachments in e-mails
 - Firewalls
 - Scan computer/removable media at regular intervals
 - Passwords
 - Encryptions
- Also accept any other correct alternative response to protect the computer from **software** access* (2)
- 6.6 6.6.1 *Any THREE functions of an operating system: ✓✓✓*
- Supply the users with a standard user interface
 - Manage computer resources such as memory, processor time and storage.
 - Supply the software developer with standard methods for aspects such as file handling, memory management, etc.
 - Process management - manage the execution of programs in memory
 - File management - by allowing users to create, save, delete, move and copy files onto secondary storage
 - Security of computer system
 - Networking capabilities (3)

- 6.6.2 *Any TWO advantages of proprietary OS over open source OS: ✓✓*
- More reliable
 - Is more widely tested
 - More user friendly/customer oriented
 - Better support available for anti-virus, firewall, anti-spam and anti-spyware
 - Is protected against changing the source code – open source changes may corrupt OS.
 - More software/applications is available for use
- (2)

- 6.6.3 *Mark allocation: ✓(not compatible) ✓(motivation)*
- Mac and Windows operating systems use different file systems and are built and operate very differently.
 - Not compatible/the various operating systems do not support software from different platforms
 - Different instruction sets
- (2)

- 6.7 6.7.1 RAID is a group of hard drives✓ with a special controller that manages multiple hard drives as one (large logical) drive.✓
- (2)

- 6.7.2
- With RAID 1 the data is mirrored✓ (on another disk whereas RAID 0 uses striping)
 - RAID 1 has high fault tolerance✓ (if access to one disk fails, the data is available on the other disk whereas with RAID 0 if any disk fails all the data is lost)

NOTE: Also accept an explanation why RAID 0 is not suitable

(2)

- 6.7.3 *Mark allocation:*
✓(example/fact) ✓(explanation/motivation of example/fact)

OR

Any TWO facts of why 'cloud' computing is beneficial for green solutions in terms of hardware:

- reduce the use of electricity as fewer hardware devices is required for onsite storage
- less resource intensive hard drives is required as the storage is taking place on the cloud
- by making use of cloud software the need for hard drives at user's station is not required
- the use of e-printing reducing the use of a printer at both nodes/it is no longer required to print documents as it is available in the cloud.

Also accept any other correct alternative answer

(2)

- 6.8 6.8.1 *Systems analyst*: analyses the requirements✓ of the client and designs a system/software✓ accordingly (2)
- 6.8.2 *Any TWO of: ✓✓*
- Network administrator/specialist
 - IT Technician
 - Webmaster/web designer
 - Computer programmer/software engineer
 - Security specialist/white-hat hacker
- NOTE: DO NOT ACCEPT:**
- database administrator
 - data capturer
 - system analyst (2)
- 6.9 *Any TWO aspects: ✓✓*
- Faster and improved data processing: (ANY ONE aspect)*
- counting of votes – faster electronically by clickers opposed to manual
 - more accurate – prevent human errors during counting process
 - using ICT as opposed to snail mail
- Improved data storage: (ANY ONE aspect)*
- large amounts of data can be stored in a small space/CD/DVD's as opposed to storage space/rooms required for storing paper votes/reports, etc.
 - duplicates (backups) can be created immediately – time consuming with paper votes
- Data communication (ANY ONE aspect)*
- Speed – information such as entry forms and results/reports can be sent much faster via the Internet using e-mails for example.
 - Professional reports can be compiled faster than handwritten reports.
 - Possible for the television audience to vote and process their votes as part of the event - not possible without ICT
- Accept any other correct and sensible answers (2)*
- 6.10 6.10.1 *Any ONE of ✓:*
- Google
 - Yahoo
 - Bing
 - Webcrawler
 - Aardvark
- Also accept any other correct/known Internet search engine (1)*

6.10.2 Any **TWO hints**✓✓ and two corresponding examples ✓✓
Mark allocation for each hint:

- Hint for search – one mark per hint
- Example applicable **to this scenario** – one mark per example

Hint	Example – accept any correct alternative examples
Use keywords related to the competition	talent competition
Make use of Boolean operators	talent AND competition
Do not use too specific search criteria - no results may be produced	Entry level for talent amongst young people
If no results were obtained - check the spelling of the words used	Talent comp <u>a</u> rtition { <i>spelling mistake made</i> }
Use advanced search features/options	Use a date range
Use synonyms	Talent contest instead of talent competition
Specific search phrase in quotes/be more specific	"talent competition"

(4)

TOTAL SECTION E: 50
GRAND TOTAL: 180